R&S®RTO6 Oscilloscope User Manual

dataTec

Mess- und Prüftechnik. Die Experten.

Ihr Ansprechpartner / Your Partner:

dataTec AG
E-Mail: info@datatec.eu
>>> www.datatec.eu

1801668702
Version 05

ROHDE&SCHWARZ
Make ideas real

This manual describes the following R&S®RTO6 models with firmware version 5.20:

- R&S®RTO64 (1802.0001K04)

© 2022 Rohde & Schwarz GmbH & Co. KG
Muehldorfstr. 15, 81671 Muenchen, Germany
Phone: +49 89 41 29 - 0
Email: info@rohde-schwarz.com
Internet: www.rohde-schwarz.com
Subject to change – data without tolerance limits is not binding.
R&S® is a registered trademark of Rohde & Schwarz GmbH & Co. KG.
The terms HDMI and HDMI High-Definition Multimedia Interface, and the HDMI Logo are trademarks or registered trademarks of HDMI Licensing Administrator, Inc. in the United States and other countries.
TEKTRONIX, TEK, and TEKPROBE are registered trademarks of Tektronix, Inc.
All other trademarks are the properties of their respective owners.

1801.6687.02 | Version 05 | R&S®RTO6

Throughout this manual, products from Rohde & Schwarz are indicated without the ® symbol, e.g. R&S®RTO6 is indicated as R&S RTO6.

Contents

1 Safety and regulatory information....11
1.1 Safety instructions....11
1.2 Labels on the product....16
1.3 Warning messages in the documentation....16
1.4 Where to find key documents on Rohde & Schwarz....17
1.5 Korea certification class A....17

2 Preface....18
2.1 Key features....18
2.2 Documentation overview....18
2.3 Options described in this document....20
2.4 Conventions used in the documentation....22

3 Getting started....23
3.1 Preparing for use....23
3.2 Instrument tour....29
3.3 Operating the instrument....41

4 Instrument setup....67
4.1 System settings....67
4.2 Option settings....71
4.3 Appearance settings....74
4.4 Display settings....90
4.5 Frontpanel settings....94
4.6 Preset setup....98
4.7 Maintenance settings....100
4.8 Save / recall....104

5 Acquisition and waveform setup....109
5.1 Basics....109
5.2 Horizontal....119
5.3 Horizontal accuracy....123
5.4 Acquire settings....126
5.5 Vertical setup....137

5.6 Probes....145
5.7 R&S RT-ZVC probe....171
5.8 Differential signals....180
5.9 Setting up the waveform....185

6 Triggers....190
6.1 Basics of triggering....190
6.2 Setting up the trigger....191
6.3 Trigger types....193
6.4 Mode / Holdoff....220
6.5 Conditioning....222
6.6 Actions on trigger....224
6.7 Qualification....226
6.8 Sequence....229
6.9 Zone trigger....232
6.10 External trigger input....236

7 Waveform analysis....239
7.1 Zoom....239
7.2 Reference waveforms....249
7.3 Mathematics....256
7.4 History....274
7.5 XY-diagram....280

8 Measurements....284
8.1 Cursor measurements....284
8.2 Automatic measurements....295
8.3 Quick measurements....368

9 Spectrum analysis....370
9.1 FFT analysis....370
9.2 Spectrum analysis (option R&S RTO6-K37)....390

10 Mask testing....394
10.1 About mask testing....394
10.2 Mask test settings....396
10.3 Mask testing on eye diagrams....408

10.4 Working with masks....411

11 Search functions....419

11.1 Overview: search definition and results....419

11.2 Search setup....420

11.3 Search gate....435

11.4 Search display presentation....438

12 Data and file management....443

12.1 Save and recall user settings....443

12.2 Save and recall waveform data and results....452

12.3 Screenshots....475

12.4 Reports....480

12.5 File selection dialog....482

13 Protocol analysis....484

13.1 Basics of protocol analysis....484

13.2 I²C (option R&S RTO6-K510)....494

13.3 SPI bus (option R&S RTO6-K510)....509

13.4 UART/RS-232/RS-422/RS-485 (option R&S RTO6-K510)....520

13.5 Audio signals (option R&S RTO6-K510)....529

13.6 Custom: Manchester / NRZ (option R&S RTO6-K510)....547

13.7 CAN and CAN FD (option R&S RTO6-K520)....578

13.8 LIN (option R&S RTO6-K520)....604

13.9 FlexRay (option R&S RTO6-K520)....618

13.10 SENT (option R&S RTO6-K520)....634

13.11 CXPI (option R&S RTO6-K520)....666

13.12 MIL-1553 (option R&S RTO6-K530)....685

13.13 ARINC 429 (option R&S RTO6-K530)....702

13.14 SpaceWire (option R&S RTO6-K530)....714

13.15 Ethernet 10BASE-T and 100BASE-TX (option R&S RTO6-K540)....726

13.16 MDIO (option R&S RTO6-K540)....741

13.17 RFFE (option R&S RTO6-K550)....754

13.18 Ethernet 100BASE-T1 (option R&S RTO6-K560)....776

13.19 Ethernet 1000BASE-T1 (option R&S RTO6-K560)....794

13.20 USB (option R&S RTO6-K570)..... 812
13.21 USB 3.1 (option R&S RTO6-K570)..... 840
13.22 USBPD (option R&S RTO6-K570)..... 861
13.23 D-PHY (option R&S RTO6-K580)..... 874
13.24 M-PHY and USB SSIC (option R&S RTO6-K580 and K570)..... 891
13.25 8b/10b (option R&S RTO6-K590)..... 921
13.26 PCIe (option R&S RTO6-K72)..... 934
13.27 DDR (option R&S RTO6-K91)..... 964

14 Mixed signal option (MSO, R&S RTO6-B1)..... 984
14.1 Digital channels and parallel buses..... 985
14.2 Display..... 994
14.3 Trigger..... 996
14.4 Measurements on digital channels..... 1009
14.5 Data export..... 1010
14.6 Mathematics..... 1011
14.7 Search..... 1011

15 Waveform generator (Option R&S RTO6-B6)..... 1013
15.1 Setup of the waveform generator..... 1013
15.2 Setup of the pattern generator..... 1032
15.3 Output..... 1035
15.4 Coupling settings..... 1038
15.5 Sync settings..... 1039
15.6 Configuring the waveform generator..... 1040

16 Pulse source and TDR/TDT analysis (options R&S RTO6-B7/K130) 1044
16.1 Pulse source (option R&S RTO6-B7)..... 1044
16.2 TDR/TDT analysis (option R&S RTO6-K130)..... 1046

17 I/Q software interface (Option R&S RTO6-K11)..... 1059
17.1 I/Q mode..... 1059
17.2 I/Q sampling setup..... 1059
17.3 I/Q data output..... 1064
17.4 NFC trigger..... 1066

18 Jitter analysis and clock data recovery 1068
18.1 Jitter measurements (options R&S RTO6-K12/K13) 1069
18.2 Advanced jitter and noise (option R&S RTO6- K133/K134) 1079
18.3 Clock data recovery 1112
19 Power analysis (option R&S RTO6-K31) 1121
19.1 Power measurement selection 1121
19.2 Overview of power measurement setup 1133
19.3 Input 1136
19.4 Control loop 1150
19.5 Power path 1164
19.6 Output 1183
20 Deembedding (option R&S RTO6-K121) 1199
20.1 Configuring the measurement setup for deembedding 1199
20.2 Deembedding settings 1200
20.3 Responses 1209
20.4 Proven probe, proven cable 1210
21 Compliance tests 1216
21.1 USB compliance test 1216
21.2 Ethernet compliance tests (options R&S RTO6-K22/K23/K24/K87/K88/K89) 1217
21.3 D-PHY compliance tests (option R&S RTO6-K26) 1220
21.4 eMMC compliance tests (option R&S RTO6-K92) 1220
21.5 PCIe compliance tests (option R&S RTO6-K81) 1221
21.6 DDR3 compliance tests (option R&S RTO6-K91) 1222
22 Network operation and remote control 1223
22.1 Operating system 1223
22.2 Setting up a network (LAN) connection 1228
22.3 Web interface 1233
22.4 Remote Desktop connection 1238
22.5 Remote control 1240
23 Maintenance and support 1255
23.1 Cleaning 1255
23.2 Contacting customer support 1255

23.3 Information for technical support....1256
23.4 Data security....1256
23.5 Transporting....1256
23.6 Storage....1257
23.7 Disposal....1257

24 Remote control commands....1258
24.1 Conventions used in remote command description....1258
24.2 Finding the appropriate command....1258
24.3 Programming examples....1259
24.4 Frequently used parameters and suffixes....1279
24.5 Common commands....1287
24.6 General remote settings....1292
24.7 Instrument setup....1296
24.8 Acquisition and setup....1324
24.9 Trigger....1388
24.10 Waveform analysis....1437
24.11 Cursor measurements....1468
24.12 Automatic measurements....1479
24.13 Spectrum analysis....1565
24.14 Mask testing....1582
24.15 Search....1603
24.16 Data management....1640
24.17 Protocols....1669
24.18 Mixed signal option (MSO, R&S RTO6-B1)....2420
24.19 Waveform generator (option R&S RTO6-B6)....2443
24.20 Pulse Source and TDR/TDT Analysis (options R&S RTO6-B7/K130)....2464
24.21 I/Q software interface (option R&S RTO6-K11)....2472
24.22 Jitter analysis and clock data recovery....2479
24.23 Power analysis (option R&S RTO6-K31)....2519
24.24 Deembedding (option R&S RTO6-K121)....2558
24.25 Maintenance....2570
24.26 Status reporting....2572
24.27 Remote trace....2578

24.28 Deprecated commands.....2581

List of commands..... 2583

Index.....2665

1 Safety and regulatory information

The product documentation helps you to use the product safely and efficiently. Follow the instructions provided here and in the Chapter 1.1, "Safety instructions", on page 11.

Intended use

The R&S RTO6 oscilloscope is designed for measurements on circuits that are only indirectly connected to the mains or not connected at all. It is not rated for any measurement category.

The product is intended for the development, production and verification of electronic components and devices in industrial, administrative, and laboratory environments. Use the product only for its designated purpose. Observe the operating conditions and performance limits stated in the data sheet.

Where do I find safety information?

Safety information is part of the product documentation. It warns you of potential dangers and gives instructions on how to prevent personal injury or damage caused by dangerous situations. Safety information is provided as follows:

- In Chapter 1.1, "Safety instructions", on page 11. The same information is provided in many languages as printed "Safety Instructions". The printed "Safety Instructions" are delivered with the product.
- Throughout the documentation, safety instructions are provided when you need to take care during setup or operation.

1.1 Safety instructions

Products from the Rohde & Schwarz group of companies are manufactured according to the highest technical standards. To use the products safely, follow the instructions provided here and in the product documentation. Keep the product documentation nearby and offer it to other users.

Use the product only for its intended use and within its performance limits. Intended use and limits are described in the product documentation such as the data sheet, manuals and the printed "Safety Instructions". If you are unsure about the appropriate use, contact Rohde & Schwarz customer service.

Using the product requires specialists or specially trained personnel. These users also need sound knowledge of at least one of the languages in which the user interfaces and the product documentation are available.

Reconfigure or adjust the product only as described in the product documentation or the data sheet. Any other modifications can affect safety and are not permitted.

Never open the casing of the product. Only service personnel authorized by Rohde & Schwarz are allowed to repair the product. If any part of the product is dam-

aged or broken, stop using the product. Contact Rohde & Schwarz customer service at https://www.rohde-schwarz.com/support.

In these safety instructions, the term "product" covers instruments (oscilloscopes), probes and their accessories.

Lifting and carrying the instrument

Check the data sheet for the maximum weight of the instrument. A single person can only carry a maximum of 18 kg safely depending on age, gender and physical condition. If your instrument is heavier than 18 kg, do not move or carry it by yourself.

Use the instrument handles to move or carry the instrument. Do not use the mounted accessories instead of the handles. Accessories are not designed to carry the weight of the instrument.

To move the instrument safely, you can use lifting or transporting equipment such as lift trucks and forklifts. Follow the instructions provided by the equipment manufacturer.

Choosing the operating site

Only use the product indoors. The product casing is not waterproof. Water that enters can electrically connect the casing with live parts, which can lead to electric shock, serious personal injury or death if you touch the casing. If Rohde & Schwarz provides accessories designed for your product, e.g. a carrying bag, you can use the product outdoors.

Unless otherwise specified in the data sheet, you can operate the product up to an altitude of 2000 m above sea level.

The product is suitable for pollution degree 2 environments where nonconductive contamination can occur. For more information on environmental conditions such as ambient temperature and humidity, see the data sheet.

Setting up the product

Always place the product on a stable, flat and level surface with the bottom of the product facing down. If the product is designed for different positions, secure the product so that it cannot fall over.

If the product has foldable feet, always fold the feet completely in or out to ensure stability. The feet can collapse if they are not folded out completely or if the product is moved without lifting it. The foldable feet are designed to carry the weight of the product, but not an extra load.

If stacking is possible, keep in mind that a stack of products can fall over and cause injury.

If you mount products in a rack, ensure that the rack has sufficient load capacity and stability. Observe the specifications of the rack manufacturer. Always install the products from the bottom shelf to the top shelf so that the rack stands securely. Secure the product so that it cannot fall off the rack.

Connecting to power and grounding

The mains power supply input of the instrument complies with overvoltage category II. It has to be connected to a fixed installation used to supply energy-consuming equipment such as household appliances and similar loads. Be aware that electrically powered products have risks, such as electric shock, fire, personal injury or even death.

Take the following measures for your safety:

- Do not use an isolating transformer to connect the instrument to the mains power supply.
- Before switching on the product, ensure that the voltage and frequency indicated on the product match the available power source. If the power adapter does not adjust automatically, set the correct value and check the rating of the fuse.
- Only use the power cable delivered with the product. It complies with country-specific safety requirements. Only insert the plug into an outlet with protective conductor terminal.
- If a product has an exchangeable fuse, its type and characteristics are indicated next to the fuse holder. Before changing the fuse, switch off the instrument and disconnect it from the power source. How to change the fuse is described in the product documentation.
- Only use intact cables and route them carefully so that they cannot be damaged. Check the power cables regularly to ensure that they are undamaged. Also ensure that nobody can trip over loose cables.
- If the product needs an external power supply, use the power supply that is delivered with the product or that is recommended in the product documentation or a power supply that conforms to the country-specific regulations.
- Ensure that you can disconnect the product from the power source at any time. Pull the power plug to disconnect the product. The power plug must be easily accessible. If the product is integrated into a system that does not meet these requirements, provide an easily accessible circuit breaker at the system level.
- Replace parts that are relevant to safety only by original parts, e.g. power cables or fuses.

Performing measurements

Take the following measures for your safety:

- To ascertain voltage-free state, use an appropriate voltage tester. Any measurement setup including an oscilloscope is not suitable for this purpose.
- The maximum input voltage on channel inputs and the external trigger input must not exceed the value specified in the data sheet.
- Observe all voltage and current ratings of the instrument, the probes, and the accessories. Exceeding the allowed voltages can lead to an electric shock. Limits and ratings are marked on the products and listed in the data sheets. Consider that the rated voltage depends on the frequency. The voltage limitation curves or values are provided in the data sheet.
- Never cause any short circuits when measuring sources with high output currents.
- Use only probes and accessories that comply with the measurement category (CAT) of your measurement task. The measurement category of the products is

defined in the data sheet. If you use other than Rohde & Schwarz accessories, make sure that they are suitable for the instrument and the measurement task.

- Set the correct attenuation factor on the instrument according to the probe being used. Otherwise, the measurement results do not reflect the actual voltage level, and you might misjudge the actual risk.
- When working with high voltages and current probes, observe the additional operating conditions specified in this safety instructions.
- The probe pins are extremely pointed and can easily penetrate clothes and the skin. Handle the probe pins with great care. To exchange a probe pin, use tweezers or pliers to avoid injuries. When transporting the accessories, always use the box supplied with the probe.
- Prevent the probe from receiving mechanical shock. Avoid putting excessive strain on the probe cable or exposing it to sharp bends. Touching a broken cable during measurements can cause injuries.
- Set up all probe connections to the instrument before applying power.

Working with hazardous voltages

Voltages higher than 30 V RMS, or 42 V peak, or 60 V DC are regarded as hazardous contact voltages. Direct contact with them can cause serious injuries.

Make sure that only electrically skilled persons use the products for measurements on hazardous contact voltages. These working conditions require special education and experience to perceive risks and to avoid hazards which electricity can create.

When working with hazardous contact voltages, use protective measures to preclude direct contact with the measurement setup:

- Do not touch exposed connections and components when power is applied.
- Switch off the test circuit while connecting and disconnecting probe leads.
- Use only insulated voltage probes, test leads and adapters.
- Make sure that the input leads fulfill the safety requirements for your measurement. The delivered input leads might have a jacket wear indicator that indicates a worn jacket by different jacket color. In this case, do not use the input lead. Replace it with a new one.
- Do not use 4 mm banana plugs without protection against contact.

Working with current probes

When working with current probes, you can measure high-frequency currents or currents that contain high-frequency components.

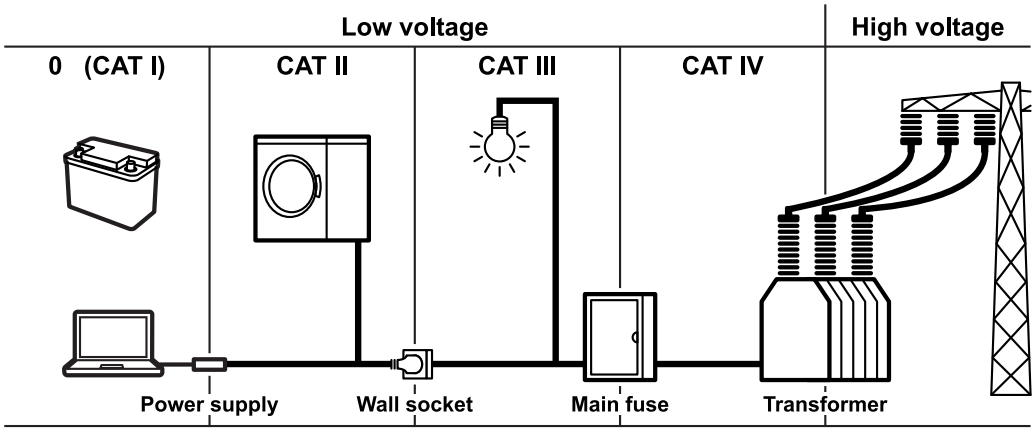
- Switch off the test circuit while connecting the probe.
- Do not attach the clamp to bare unisolated conductors. To avoid injury from a short circuit, measure at a location on an insulated wire where the insulation is sufficient for the circuit voltage.
- Connect the probe only to the secondary side of a breaker. With this measure, you avoid injury, if a short circuit occurs.
- The following effects can cause burns and fire or damage to the measurement site:

– Eddy current loss can cause heating of the sensor head.
– Dielectric heating can cause heating of cord insulation and other materials.

Measurement categories

IEC 61010-2-030 defines measurement categories that rate instruments on their ability to resist short transient overvoltages that occur in addition to the working voltage. Use the measurement setup only in electrical environments for which they are rated.

- 0 - Instruments without rated measurement category
  For measurements performed on circuits not directly connected to mains, for example, electronics, circuits powered by batteries, and specially protected secondary circuits. This measurement category is also known as CAT I.
- CAT II:
  For measurements performed on circuits directly connected to the low-voltage installation by a standard socket outlet, for example, household appliances and portable tools.
- CAT III:
  For measurements performed in the building installation, such as junction boxes, circuit breakers, distribution boards, and equipment with permanent connection to the fixed installation.
- CAT IV:
  For measurements performed at the source of the low-voltage installation, such as electricity meters and primary overcurrent protection devices.



Cleaning the product

Use a dry, lint-free cloth to clean the product. When cleaning, keep in mind that the casing is not waterproof. Do not use liquid cleaning agents.

Meaning of safety labels

Safety labels on the product warn against potential hazards.

	Potential hazard Read the product documentation to avoid personal injury or product damage.
	Heavy product Be careful when lifting, moving or carrying the product. Carrying the product requires at least two people or transport equipment.
	Electrical hazard Indicates live parts. Risk of electric shock, fire, personal injury or even death.
	Protective conductor terminal Connect this terminal to a grounded external conductor or to protective ground. This protects you against electric shock should an electric problem occur.

1.2 Labels on the product

Labels on the casing inform about:

- Personal safety, see "Meaning of safety labels" on page 15
- Product and environment safety, see Table 1-1
- Identification of the product

Table 1-1: Labels regarding product and environment safety

	Chassis grounding terminal
	Take care when handling electrostatic sensitive devices.
	Labeling in line with EN 50419 for disposal of electrical and electronic equipment after the product has come to the end of its service life. For more information, see "Disposing electrical and electronic equipment" on page 1257.

1.3 Warning messages in the documentation

A warning message points out a risk or danger that you need to be aware of. The signal word indicates the severity of the safety hazard and how likely it will occur if you do not follow the safety precautions.

WARNING

Potentially hazardous situation. Could result in death or serious injury if not avoided.

CAUTION

Potentially hazardous situation. Could result in minor or moderate injury if not avoided.

NOTICE

Potential risks of damage. Could result in damage to the supported product or to other property.

1.4 Where to find key documents on Rohde & Schwarz

Certificates issued to Rohde & Schwarz that are relevant for your country are provided at www.rohde-schwarz.com/key-documents, e.g. concerning:

- Quality management
- Environmental management
- Information security management
- Accreditations

1.5 Korea certification class A

이 기기는 업무용(A급) 전자파 적합기기로서 판매자 또는 사용자는 이 점을 주의하시기 바라며, 가정외의 지역에서 사용하는 것을 목적으로 합니다.

2 Preface

2.1 Key features

The R&S RTO6 oscilloscope provides fast signal acquisition and analysis:

- 1 million waveforms per second waveform acquisition rate
- Highly accurate digital trigger system
- Very low noise floor
- Precise measurements due to single-core A/D converter
- High measurement speed, even for complex analysis functions
- High-quality line of probes

For a detailed specification refer to the data sheet.

The R&S RTO6 oscilloscope brings various benefits in your daily work. It helps you to:

- Find signal anomalies quickly with unparalleled update rate
- Capture all signal details with excellent signal integrity
- See more with up to 16-bit resolution
- Find complex signal details with advanced trigger capabilities
- Measure more with deep and responsive memory

The large 15,6” touchscreen and the streamlined user interface of R&S RTO6 oscilloscopes help you quickly solve circuit issues from simple to complex.

2.2 Documentation overview

This section provides an overview of the R&S RTO6 user documentation.

2.2.1 Manuals and instrument help

You find the manuals on the product page at:

www.rohde-schwarz.com/manual/rto6

Getting started manual

Introduces the R&S RTO6 and describes how to set up and start working with the instrument, and describes basic operations. A printed English version is included in the delivery. Editions in other languages are available on the product website.

Instrument help

The help offers quick, context-sensitive access to the complete information for the firmware basic functionality and applications.

User manual (Instrument)

Describes all instrument functions in detail. It also provides an introduction to remote control, a complete description of the remote control commands with programming examples, and information on maintenance and instrument interfaces. Includes the contents of the getting started manual.

The *online version* of the user manual provides the complete contents for immediate display on the internet.

Manuals for compliance test options

For compliance test options, extra test procedure manuals are available. Test fixtures are described in printed manuals, which are delivered with the fixture.

- The following test procedure manuals are available:
  - USB 2.0 Compliance Test Procedures
  - Ethernet Compliance Tests Procedures
  - MIPI D-PHY Compliance Tests Procedures
  - eMMC Compliance Tests Procedures
  - PCIe Compliance Tests Procedures
  - DDR3 Compliance Tests Procedures
  - ScopeSuite Automation
- The following test fixture manuals are available:
  - R&S RT-ZF1 USB 2.0 Compliance Test Fixture Set
  - R&S RT-ZF2 Ethernet Compliance Test Fixture Set
  - R&S RT-ZF3 Frequency Converter Board (100BASE-T1)
  - R&S RT-ZF4 10BASE-Te Test Fixture
  - R&S RT-ZF5 Ethernet Probing Fixture
  - R&S RT-ZF6 Frequency Converter Board (1000BASE-T1)
  - R&S RT-ZF7 Automotive Ethernet T&D Fixture
  - R&S RT-ZF7A and R&S RT-ZF7P SMA Adaptors
  - R&S RT-ZF8 Automotive Ethernet Compliance Test Fixture

Safety instructions

Provides safety information in many languages. The printed document is delivered with the product.

Instrument security procedures manual

Deals with security issues when working with the R&S RTO6 in secure areas. It is available for download on the internet.

Service Manual

Describes the performance test for checking the rated specifications, module replacement, firmware update, troubleshooting and fault elimination, and contains mechanical drawings and spare part lists. The service manual is available for registered users on

the global Rohde & Schwarz information system (GLORIS, https://gloris.rohde-schwarz.com).

2.2.2 Data sheet and brochure

The data sheet contains the technical specifications of the R&S RTO6. It also lists the firmware applications and their order numbers, and optional accessories. The brochure provides an overview of the instrument and deals with the specific characteristics.

www.rohde-schwarz.com/brochure-datasheet/rto6

2.2.3 Release notes, open source acknowledgment

The release notes list new features, improvements and known issues of the current firmware version, and describe the firmware installation. The open source acknowledgment document provides verbatim license texts of the used open source software. It can also be read directly on the instrument.

www.rohde-schwarz.com/firmware/rto6

2.2.4 Application notes, application cards, videos

These documents deal with special applications or background information on particular topics.

- Application notes
- Media Center

2.3 Options described in this document

In addition to the base unit, the following options are described in this documentation:

Type	Designation	Order No.
R&S RTO6-B1	MSO	1801.6741.02
R&S RTO6-B6	Waveform and pattern generator	1801.6758.02
R&S RTO6-B7	Pulse source	1801.6764.02
R&S RTO6-K11	I/Q software interface	1801.6812.02
R&S RTO6-K12	Jitter analysis	1801.6829.02
R&S RTO6-K13	Clock data recovery	1801.6835.02
R&S RTO6-K133	Advanced jitter analysis	1801.6906.02
R&S RTO6-K134	Advanced noise analysis	1801.7677.02
R&S RTO6-K37	Spectrum analysis	1801.6870.02

Type	Designation	Order No.
R&S RTO6-K31	Power analysis	1801.6858.02
R&S RTO6-K121	Deembedding base option	1801.6887.02
R&S RTO6-K130	TDR/TDT analysis (requires pulse source option R&S RTO6-B7) For R&S RTO6 with at least 2 GHz bandwidth	1801.6893.02
R&S RTO6-K500	Bus analysis for I2C, SPI, UART, CAN/CAN-FD, LIN, SENT Ethernet(10BASE-T/100BASE-Tx), RFFE, Automotive Ethernet (100BASE-T1/1000BASE-T1) Requires corresponding protocol decoding option.	1801.6864.02
R&S RTO6-510	Triggering and decoding Low-speed Serial Buses: I2C , SPI, UART/RS-232/RS-422/RS-485, Audio I2S, Custom Manchester / NRZ	1801.7019.02
R&S RTO6-520	Triggering and decoding Automotive Protocols: LIN, CAN / CAN-FD, FlexRay, SENT, CXPI	1801.7025.02
R&S RTO6-530	Triggering and decoding Aerospace Protocols: MIL-STD-1553, ARINC 429, SpaceWire	1801.7031.02
R&S RTO6-540	Triggering and decoding Ethernet Protocols: 10BASE-T / 100BASE-Tx, MDIO	1801.7048.02
R&S RTO6-550	Triggering and decoding MIPI RFFE	1801.7054.02
R&S RTO6-560	Triggering and decoding Automotive Ethernet: 100BASE-T1, 1000BASE-T1	1801.7060.02
R&S RTO6-570	Triggering and decoding USB Protocols: USB 1.0/1.1/2.0, USB 3.1 Gen 1, USB Power Delivery (USB-PD), USB 3.1 SSIC	1801.7077.02
R&S RTO6-580	Triggering and decoding MIPI M-PHY, D-PHY	1801.7083.02
R&S RTO6-590	Triggering and decoding 8b10b, PCI Express 1.1 / 2.0	1801.7090.02

Type	Designation	Order No.
R&S RTO6--TDBDL	Trigger and decode bundle. Includes options: R&S RTO6- -K500,-K510, -K520, -K530, -K540, -K550, -K560, -K570, -K580, -K590	1801.7725.02
R&S RTO6-K91	DDR3/DDR3L/LPDDR3 debug & compliance test. Decode and debug is described in this user manual. The compliance test is described in a separate manual.	1801.6993.02

2.4 Conventions used in the documentation

2.4.1 Typographical conventions

The following text markers are used throughout this documentation:

Convention	Description
"Graphical user interface elements"	All names of graphical user interface elements on the screen, such as dialog boxes, menus, options, buttons, and softkeys are enclosed by quotation marks.
[Keys]	Key and knob names are enclosed by square brackets.
`Filenames, commands, program code`	Filenames, commands, coding samples and screen output are distinguished by their font.
Input	Input to be entered by the user is displayed in italics.
Links	Links that you can click are displayed in blue font.
"References"	References to other parts of the documentation are enclosed by quotation marks.

2.4.2 Conventions for procedure descriptions

When operating the instrument, several alternative methods may be available to perform the same task. In this case, the procedure using the touchscreen is described. Any elements that can be activated by touching can also be clicked using an additionally connected mouse. The alternative procedure using the keys on the instrument or the on-screen keyboard is only described if it deviates from the standard operating procedures.

The term "select" may refer to any of the described methods, i.e. using a finger on the touchscreen, a mouse pointer in the display, or a key on the instrument or on a keyboard.

3 Getting started

Note: the following chapters are identical to those in the R&S RTO6 Getting Started manual for R&S RTO6 instruments.

- Preparing for use.....23
- Instrument tour.....29
- Operating the instrument.....41

3.1 Preparing for use

Here, you can find basic information about setting up the instrument for the first time or when changing the operating site.

3.1.1 Lifting and carrying

See: "Lifting and carrying the instrument" on page 12.

3.1.2 Unpacking and checking

1. Unpack the product carefully.
2. Retain the original packing material. Use it when transporting or shipping the product later.
3. Using the delivery notes, check the equipment for completeness.
4. Check the equipment for damage.

   If the delivery is incomplete or equipment is damaged, contact Rohde & Schwarz.

3.1.3 Choosing the operating site

Specific operating conditions ensure proper operation and avoid damage to the product and connected devices. For information on environmental conditions such as ambient temperature and humidity, see the data sheet.

See also "Choosing the operating site" on page 12.

Electromagnetic compatibility classes

The electromagnetic compatibility (EMC) class indicates where you can operate the product. The EMC class of the product is given in the data sheet.

- Class B equipment is suitable for use in:
  - Residential environments

  - Environments that are directly connected to a low-voltage supply network that supplies residential buildings
- Class A equipment is intended for use in industrial environments. It can cause radio disturbances in residential environments due to possible conducted and radiated disturbances. It is therefore not suitable for class B environments.
  If class A equipment causes radio disturbances, take appropriate measures to eliminate them.

3.1.4 Setting up the product

When setting up the instrument, follow the safety instructions:

- "Setting up the product" on page 12
- "Intended use" on page 11

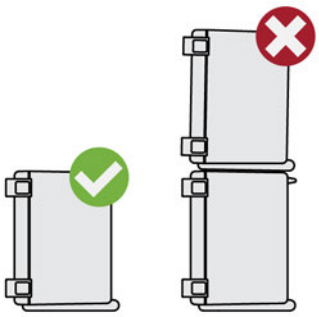
3.1.4.1 Placing the product on a bench top

For standalone operation, place the instrument on a horizontal bench with even, flat surface. The instrument can be used in horizontal position, standing on its feet, or with the support feet on the bottom extended.

To place the product on a bench top

1. Place the product on a stable, flat and level surface.
2. **CAUTION!** The top surface of the product is too small for stacking. If you stack another product on top of the product, the stack can fall over and cause injury.

   If you want to save space, mount several products in a rack.



3. **CAUTION!** Foldable feet can collapse. See "Setting up the product" on page 12.

   Always fold the feet completely in or out. With folded-out feet, do not place anything on top or underneath.
4. **NOTICE!** Overheating can damage the product.

   Prevent overheating as follows:

   - Keep a minimum distance of 10 cm between the fan openings of the product and any object in the vicinity.

- Do not place the product next to heat-generating equipment such as radiators or other products.

3.1.4.2 Mounting the product in a rack

The instrument can be installed in a rack using a rack adapter kit. The order number is given in the data sheet. The installation instructions are part of the adapter kit.

To prepare the rack

1. Observe the requirements and instructions in "Setting up the product" on page 12.
2. **NOTICE!** Insufficient airflow can cause overheating and damage the product.
   Design and implement an efficient ventilation concept for the rack.

To mount the R&S RTO6 in a rack

1. Use an adapter kit that fits the dimensions of the R&S RTO6 to prepare the instrument for rack mounting. For information on the dimensions, see data sheet.
   a) Order the rack adapter kit designed for the R&S RTO6. For the order number, see data sheet.
   b) Mount the adapter kit. Follow the assembly instructions provided with the adapter kit.
2. Push the product onto the shelf until the rack brackets fit closely to the rack.
3. Tighten all screws at the rack brackets with a tightening torque of 1.2 Nm to secure the product at the rack.

To unmount the product from a rack

1. Loosen the screws at the rack brackets.
2. Remove the product from the rack.
3. If placing the product on a bench top again, unmount the adapter kit from the product. Follow the instructions provided with the adapter kit.

3.1.5 Considerations for test setup

Observe safety instructions, see "Performing measurements" on page 13.

Cable selection and electromagnetic interference (EMI)

Electromagnetic interference (EMI) can affect the measurement results.

To suppress electromagnetic radiation during operation:

- Use high-quality shielded cables, for example, double-shielded RF and LAN cables.
- Always terminate open cable ends.
- Ensure that connected external devices comply with EMC regulations.

Measuring accessories

Use only probes and measuring accessories that comply with IEC 61010-031.

Signal input and output levels

Information on signal levels is provided in the data sheet. Keep the signal levels within the specified ranges to avoid damage to the product and connected devices.

Preventing electrostatic discharge (ESD)

Electrostatic discharge is most likely to occur when you connect or disconnect a DUT.

► **NOTICE!** Electrostatic discharge can damage the electronic components of the product and the device under test (DUT).

Ground yourself to prevent electrostatic discharge damage:

a) Use a wrist strap and cord to connect yourself to ground.
b) Use a conductive floor mat and heel strap combination.

During operation, if the firmware observes a serious unexpected disturbance (e.g. due to ESD), it resets some hardware components and initiates a new alignment to ensure proper instrument functioning. Then it restores the user settings to the state before the disturbance.

3.1.6 Connecting to power

For safety information, see "Connecting to power and grounding" on page 13.

The R&S RTO6 can be used with different AC power voltages and adapts itself automatically to it.

The nominal ranges are:

- 100 V to 240 V AC at 50 Hz to 60 Hz and 400 Hz, with maximal 10% voltage fluctuation on line
- max. 6.2 A to 4.1 A

1. Plug the AC power cable into the AC power connector on the rear panel of the product. Only use the AC power cable delivered with the product.
2. Plug the AC power cable into a power outlet with ground contact.

   The required ratings are listed next to the AC power connector and in the data sheet.

3.1.7 Switching on or off

The instrument is switched on or off with the power switch and the [Power] key. The [Power] key is located in the bottom left corner of the front panel. The power switch is located at the rear panel of the instrument.

Table 3-1: Overview of power states

Status	LED	Power switch
Off	(unlighted)	[0] (off)
Standby	yellow	[I] (on)
Ready	green	[I] (on)

To switch on the product

The product is off but connected to power.

1. Set the switch on the power supply to position [I].

   The power key on the front panel lights up.

2. Press the [Power] key on the front panel.

   The instrument performs a system check, boots the operating system, and then starts the R&S RTO6 firmware.
   The [Power] key turns green and the illuminated keys on the front panel light up. If the previous session was terminated regularly, the oscilloscope uses the last settings.

Before you start measurements, be sure to comply with the warm-up phase specified in the data sheet.

To shut down the product

The product is in the ready state.

► Press the [Power] key.

All current settings are saved, and the software shuts down. The [Power] key turns yellow. The standby power supplies only the power switch circuits.

The ⏻ "Power" icon in the "Menu" shuts down only the firmware application. To shut down the instrument completely, use the [Power] key.

To disconnect from power

The product is in the standby state.

1. **NOTICE!** Risk of data loss. If you disconnect the product from power when it is in the ready state, you can lose settings and data. Shut it down first.

   Set the switch on the power supply to position [0].

   The LED of the [Power] key is switched off.

2. Disconnect the product from the power source.

3.1.8 Connecting external devices

The following interfaces for external devices are provided:

- USB connectors at the front and rear panel of the instrument
- Monitor connectors HDMI and DisplayPort at the rear panel of the instrument

3.1.8.1 Connecting USB devices

You can connect USB flash drives to save screenshots and measurement results, and keyboard and/or mouse to simplify the operation and the entry of data. You can connect or disconnect all USB devices during operation of the instrument. The number of USB connectors can be increased by using USB hubs.

USB devices are plug&play, the operating system automatically uses the suitable device driver. If the operating system does not find a suitable driver, it prompts you to specify a directory that contains the driver software. If the driver software is on a storage media, connect the appropriate drive to the instrument before proceeding. If the instrument is integrated in a network, you can also install driver data stored in a network directory.

When a USB device is disconnected from the R&S RTO6, Windows immediately detects the change in hardware configuration and deactivates the corresponding driver.

The properties of external USB devices are configured in the operating system, not in the instrument's firmware. We recommended using mouse and keyboard to access and modify the settings of the Windows operating system.

To access Windows, press the Windows key on the external keyboard, or select "Menu" > "Minimize Application" on the R&S RTO6 menu.

Connecting a keyboard or mouse

1. Connect the keyboard or mouse.

   The device is detected automatically. The default input language of the keyboard is English – US.

2. To configure the keyboard properties, press the Windows key on the external keyboard.
3. Tap the "Find" icon (magnifier) on the Windows taskbar.
4. To configure the keyboard:
   a) Type *keybord*.
   b) Select "Edit language and keyboard options".
5. To configure the mouse:
   a) Type *mouse*.
   b) Select "Mouse settings".

3.1.8.2 Connecting external monitors

You can connect one or two external monitors or projectors to the R&S RTO6. The following connectors are available:

- "HDMI" on page 33

- "DisplayPort" on page 33

1. Ensure that the monitors and the R&S RTO6 are connected to a ground contact. Otherwise the instrument can be damaged.
2. Check the input type of the monitors. Make sure to select the correct cable. If the connectors do not match, use an appropriate adapter.
3. You can configure the displays directly in the instrument's setup. Open "Menu" > "Settings" > "Display".
4. Select the "Monitors" tab.
5. Select how to display the screen. Select "Extend" to arrange dialogs and result boxes on different displays.
6. To move the main firmware window to an external monitor, press F11 on the keyboard.

   Now the firmware shows a soft frontpanel together with the usual display in a window. You can move this window to an external monitor.
7. To access Windows display settings, tap "Additional settings".

The touchscreen of the R&S RTO6 has a screen resolution of 1920x1080 pixels pixel. Many external monitors have a higher screen resolution. If the screen resolution of the monitor is set higher than the instrument's resolution, the firmware window uses a 1920x1080 pixels area of the monitor display. For full screen display, adjust the monitor's screen resolution using "Additional display settings".

3.2 Instrument tour

This chapter describes the front and rear panels of the instrument including all function keys and connectors.

3.2.1 Front panel

The front panel of the R&S RTO6 is shown in Figure 3-1. The function keys are grouped in functional blocks to the right of the touchscreen. Below the screen, various connectors are located.

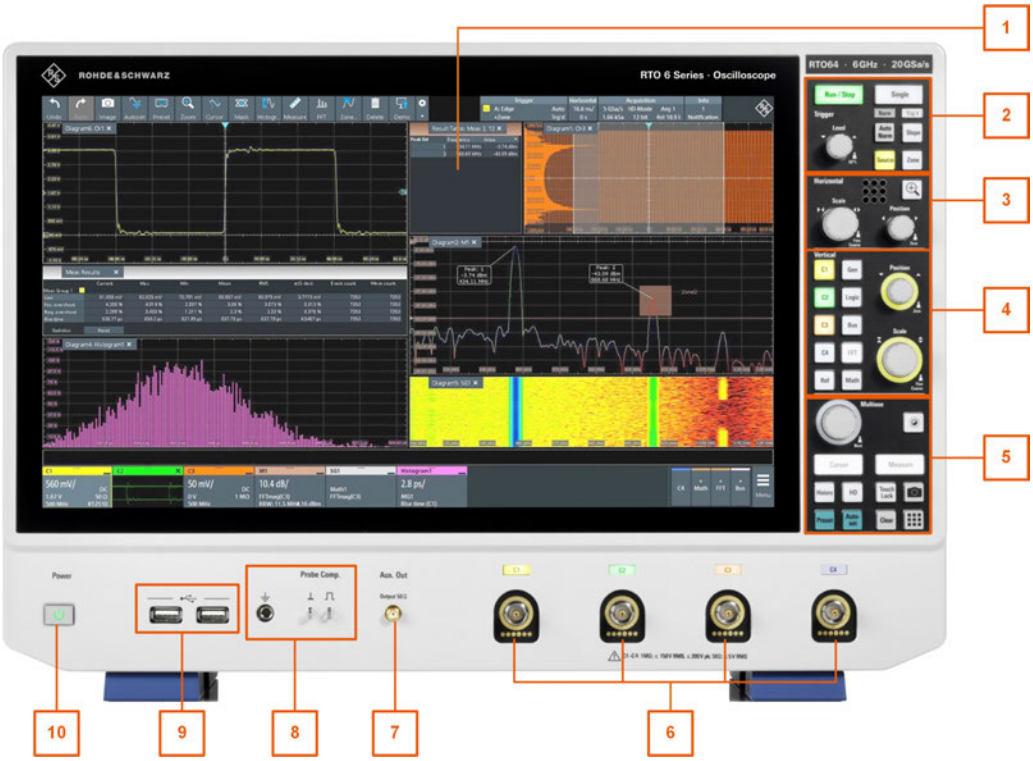


Figure 3-1: Front panel of R&S RTO6

1 = Touchscreen
2 = [Trigger] controls
3 = [Horizontal] controls
4 = [Vertical] controls
5 = [Analysis] keys
6 = Input channels
7 = [Aux Out] connector
8 = Connectors for probe compensation and grounding
9 = USB connectors
10 = [Power] key

3.2.1.1 Input connectors

The R&S RTO6 has four channel inputs to connect the input signals using active and passive probes.

The input connectors are provided with a special Rohde & Schwarz active probe interface, and they are BNC compatible. Thus, the instrument can automatically detect passive probes with standard BNC connector and active Rohde & Schwarz probes having the Rohde & Schwarz probe interface.

The input impedance is selectable, the values are 50 Ω and 1 MΩ.

The maximum input voltage is 200 V peak, 150 V RMS at 1 MΩ input impedance and 5 V RMS at 50 Ω input impedance.

3.2.1.2 Other front panel connectors

Besides the input connectors, the instrument has USB connectors and probe compensation connectors at the front panel.

[USB]
Two USB type A connectors that comply with standard USB 2.0. They are used to connect devices like keyboard, mouse and USB flash drive.

Probe Compensation
Probe compensation terminal to support adjustment of passive probes to the oscilloscope channel.

⏚	Protective earth conductor, 4 mm banana jack to connect ground of DUT and test fixtures, and wrist strap.
⎍	Square wave signal for probe compensation, 1 kHz and 1 V_{pp}.
⊥	Ground connector for probes.

[Aux Out]
Output of the internal calibration signal, if the signal is configured to external destination.

3.2.2 Rear panel

The rear panel of the R&S RTO6 is shown in Figure 3-2.

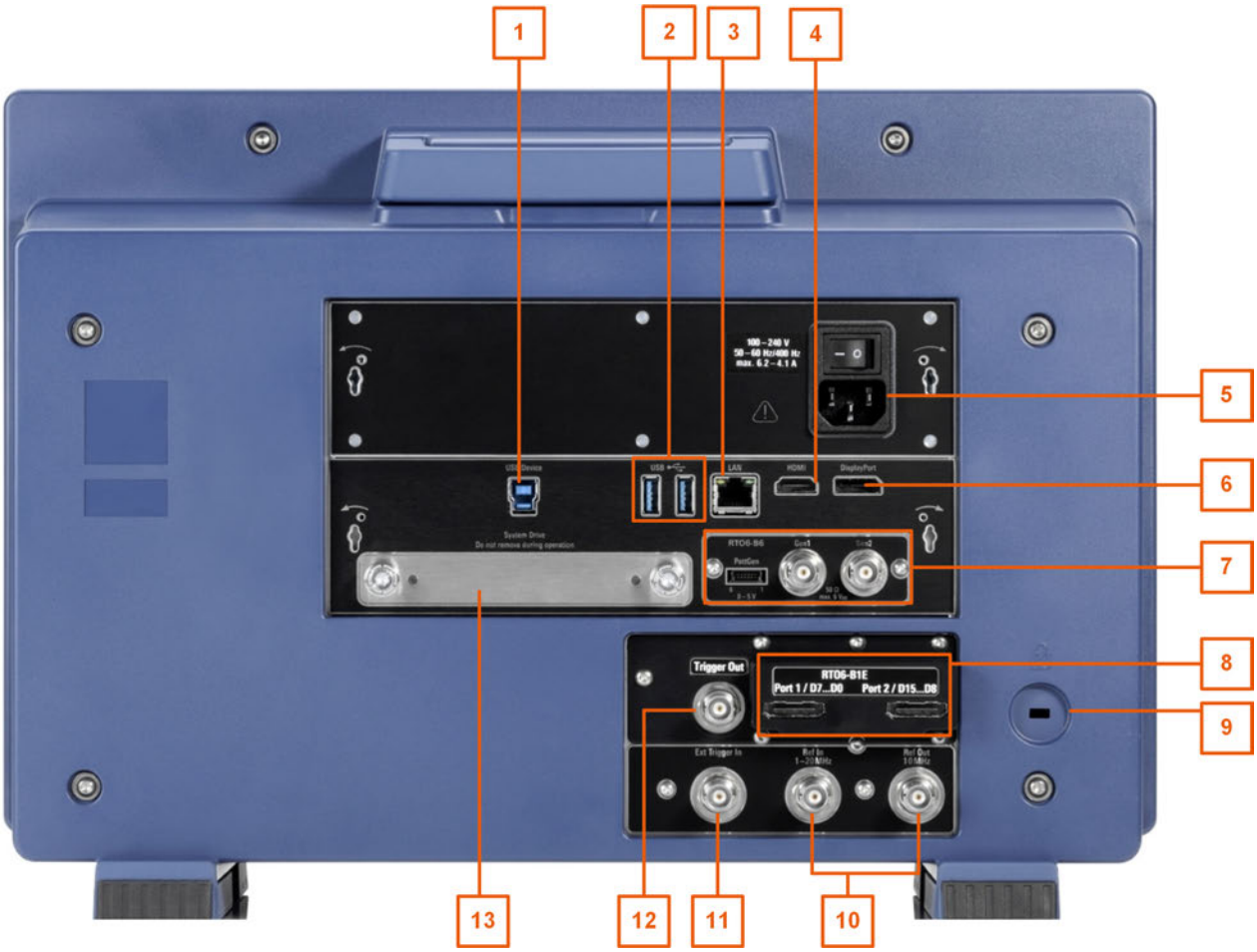


Figure 3-2: Rear panel of R&S RTO6

1 = USB device connector, type B
2 = USB connectors, type A
3 = LAN connector
4 = HDMI connector for external monitor
5 = AC power supply connector and main power switch
6 = DisplayPort connector
7 = Option slot for R&S RTO6-B6 (waveform generator, shown in figure) or R&S RTO6-B7 (pulse source)
8 = Option slot for R&S RTO6-B1 (MSO, shown in figure) or R&S RTO6-B1E (for R&S RT-ZVC), or R&S RTO6-B10 (GPIB)
9 = Kensington lock slot to secure the instrument against theft
10 = Ref In, Ref Out: reference input, and output of the OCXO reference signal
11 = External trigger input
12 = External trigger output
13 = Slot for optional exchangeable solid state disk (option R&S RTO6-B19)

AC power supply connector and main power switch

Connection to the AC power line. The R&S RTO6 can be used with different AC power voltages and adapts itself automatically to it. The nominal voltage and frequency ranges are displayed on the rear panel and quoted in the data sheet.

If grounding is *not* ensured by the mains system, ground the oscilloscope using the protective earth conductor on the front panel and an appropriate cable.

The AC main power switch also interrupts the power supply of the OCXO.

When you power up the instrument, be sure to comply with the warm-up phase specified in the data sheet before you start measurements.

See also: Chapter 3.1.6, "Connecting to power", on page 26.

USB Device
USB 3.0 interface of type B (device USB), to be used for remote control of the instrument.

USB
Two USB type A connectors that comply with standard USB 3.1 gen 1. They are used to connect devices like keyboard, mouse and flash drive to store and reload instrument settings and measurement data.

Note: Electromagnetic interference (EMI) can affect the measurement results. To avoid any impact, do not use USB connecting cables exceeding 1 m.

LAN
8-pin RJ-45 connector used to connect the instrument to a Local Area Network (LAN). It supports up to 1000 Mbit/s (10/100/1000BASE-T Ethernet).

HDMI
HDMI Version 2.0 connector for an external monitor or projector. The monitor shows the complete content of the instrument's screen.

See also: Chapter 3.1.8.2, "Connecting external monitors", on page 28.

DisplayPort
DisplayPort connector for an external monitor or projector. It supports DisplayPort++ version 1.3.

Ext Trigger In
The BNC connector for external trigger input is used to control the measurement by an external signal. The input impedance can be selected in the trigger configuration, the values are 50 Ω and 1 MΩ. The trigger level can be set from -5 V to 5 V. The maximum input voltage is 30 V RMS at 1 MΩ input impedance and 7 V RMS at 50 Ω input impedance.

Trigger Out
The BNC connector for external trigger output is used to provide the internal trigger signal of the oscilloscope to trigger other instruments for synchronized measurements.

When a trigger occurs, the R&S RTO6 creates a pulse of 5 V with a source impedance of 50 Ω and delivers it to the external trigger output. The instrument can also send the pulse on mask test violation or violation of measurement limits and margins.

If the connector is terminated with 50 Ω, the signal level is 2.5 V (50 mA). With 1 MΩ termination, the level is 5 V. A short-circuit of the connector to ground creates current of 100 mA.

To enable the trigger out signal, select the "Trigger out" action for the trigger or test. You can adjust polarity and length of the pulse. The default is a positive pulse of 100 ns. The delay is 800 ns.

Mixed signal option R&S RTO6-B1 / digital extension port R&S RTO6-B1E
The mixed signal option R&S RTO6-B1 provides logic analizer functionality and 16 digital channels. It also can connect the R&S RT-ZVC multi-channel power probe. The connectors are used to connect two logical probes with 8 digital channels each (D0 to D7 and D8 to D15), or two flat interface cables of R&S RT-ZVC.

The digital extension port R&S RTO6-B1E provides the connectors for the R&S RT-ZVC multi-channel power probe without MSO. The connectors are used to connect two flat interface cables of R&S RT-ZVC.

One of the options can be installed in the option slot at the rear panel.

The maximum input voltage is 40 V peak at 100 kΩ input impedance. The maximum input frequency for a signal with the minimum input voltage swing of 500 mV (V_{pp}) is 400 MHz. For detailed specifications, refer to the data sheet.

Waveform generator option R&S RTO6-B6
The waveform generator generates various function and arbitrary waveforms, sweeps, and parallel patterns. For detailed specifications, refer to the data sheet.

The option can be installed in the option slot at the rear panel.

The module provides the following connectors:

[Gen1, Gen2]	BNC connectors
[PattGen]	Connector for the pattern generator

Pulse source option R&S RTO6-B7
The pulse source outputs a symmetrical differential pulse signal. For detailed specifications, refer to the data sheet.

The option can be installed in the option slot at the rear panel.

The module has four connectors.
- Out, $\overline{\text{Out}}$: 2.92 mm connectors (K type) for pulse signal output
- Ref, $\overline{\text{Ref}}$: 2.92 mm connectors (K type) for reference signal output

GBIP option R&S RTO6-B10
The GBIP option provides a GBIP interface and connector for remote control. For detailed specifications, refer to the data sheet.

3.2.3 Keys and controls

3.2.3.1 Power key

The [Power] key is located on the lower left corner of the front panel. It starts up and shuts down the instrument's software.

The light of the key shows the instrument state, see Chapter 3.1.7, "Switching on or off", on page 26.

3.2.3.2 Trigger controls

The keys and knob in the Trigger functional block adjust the trigger and start or stop acquisition.

[Level]
The rotary knob sets the trigger level for all trigger types. Turn clockwise to move up the trigger level. Press the knob to set the trigger level to 50% of the signal amplitude. If the selected trigger type requires two trigger levels - upper and lower level - press the knob to toggle between the two levels.

[Auto Norm]
Toggles the trigger mode between "Auto" and "Normal". The current setting is shown on the trigger label.

Auto	The instrument triggers repeatedly after a time interval if the trigger conditions are not fulfilled.
Norm	The instrument acquires a waveform only if a trigger occurs. The "Norm" LED lights up in green.

Remote command:
`TRIGger<m>:MODE` on page 1430

[Source]
Opens a dialog box where you can select the trigger source. Press the key again to switch the source. The key lights up in the color of the selected trigger source.

Remote command:
`TRIGger<m>:SOURce[:SELect]` on page 1389

[Slope]
Toggles the trigger slope or trigger polarity, dependent on the trigger type. The current setting is shown on the trigger label.

[Run Stop]
Starts and stops the continuous acquisition. The [Run Stop] key lights green when the acquisition is running. When the acquisition is stopped, the key lights red.

Remote command:
`RUN` on page 1325
`STOP` on page 1326

[Single]
Starts a defined number of acquisitions. The [Single] key lights green when the acquisition is running. When the acquisition is stopped, the key lights red.

To set the number of acquisitions, select "Menu" > "Acquire" > "Setup" tab, and set "N-single count".

Remote command:
`SINGle` on page 1325

[Zone]
Opens the overlay menu to configure the zone trigger. A zone trigger combines the trigger condition with the intersection or non-intersection of one or more zones or masks.

Remote command:
`TRIGger<m>:ZONE:EXPRession[:DEFine]` on page 1433

3.2.3.3 Horizontal controls

The rotary knobs in the Horizontal functional block adjust the horizontal parameters. These settings are effective for all channel waveforms. In addition, a [Zoom] is available.

[Position]
The rotary knob changes the horizontal position of the waveform or the position of the reference point on the screen. To set the value to zero, press the knob. The current value is shown in the input box in the upper right corner of the screen.

You can select if the knob changes the position or the reference point in "Menu" > "Settings" > "Frontpanel" > "Knobs".

"Horizontal position" defines the time distance of the reference point from the zero point of the diagram. Turn clockwise to move the waveform to the right.

"Reference point" defines the position of the reference point on the screen. Turn clockwise to move it to the right. The reference point marks the rescaling center of the time scale. If you modify the time scale, the reference point remains fixed on the screen, and the scale is stretched or compressed to both sides of the reference point.

Remote command:
`TIMebase:HORizontal:POSition` on page 1327
`TIMebase:REFerence` on page 1327

[Scale]
The rotary knob adjusts the time scale for all signals. The time scale is also known as timebase.

Turn clockwise to stretch the waveforms. Doing so, the scale value *time/div* decreases.

Press the knob to toggle between coarse and fine scale adjustment.

[Zoom]
Displays a zoom diagram for the active diagram. The key is illuminated if at least one zoom is active. If you press the key while the zoom function is on, the "Zoom" dialog box opens, where you can configure several zoom areas for detailed signal observation.

3.2.3.4 Vertical controls

The keys and knobs in the Vertical functional block select a signal and adjust the vertical scale and position of the selected signal.

[C<n>]
Turns on and selects a channel. If the channel is active, the key lights up in the corresponding channel color.

The effect of the keypress depends on state of the channel:
- If channel is off: turns on the channel and selects it.
- If the channel is on and in focus (selected): opens the corresponding channel dialog.
- If the channel is on, but not in focus (not selected): selects the channel waveform.
- If the channel is selected, and the dialog is open: turns off the channel, and closes the dialog.

The vertical rotary knobs are focused on the selected waveform. They are illuminated in the color of the selected waveform.

Remote command:
`CHANnel<m>:STATe` on page 1337

[Ref]
Opens the "Reference" dialog box, where you can configure and display reference waveforms. Press the key repeatedly to switch the reference waveform.

If a reference waveform is selected, the vertical rotary knobs are illuminated in white or light gray (default colors), depending on the selected waveform.

[Gen]
Opens the "Waveform Generator" dialog, if option R&S RTO6-B6 is installed.

The waveform generator can generate various function and arbitrary waveforms, sweeps, and parallel patterns. For detailed specifications, refer to the data sheet.

[Logic]
Opens the "Logic" dialog to configure and enable the logic buses L1 to L4. A logic bus (or parallel bus) has up to 16 logic (digital) channels. If logic buses are active, pressing the key toggles these buses.

The key lights up if at least one logic bus is enabled.

[Bus]
Opens and closes the "Protocol" tab in the "Apps" dialog, where you can select the required serial protocol and proceed to the protocol setup. The key lights up if the decoding of a serial bus is active.

If serial buses are active, pressing the key toggles these buses when the "Apps" dialog or the "Serial Bus" dialog is open.

[FFT]
Opens and closes the "Math" dialog with FFT settings. The key lights up if an FFT is active.

[Math]
Opens the "Math" dialog, where you can configure the calculation of mathematical waveforms using various mathematic operations on other waveforms. Press the key repeatedly to toggle the selected math waveform. If no math waveform is active, the key closes the dialog.

If a math waveform is selected, the vertical rotary knobs are illuminated in light brown (default color), the brightness of the color depends on the selected waveform.

[Position]
The vertical [Position] knob changes the vertical offset or the position of the selected waveform. The horizontal axis and the selected waveform are moved vertically. You can select whether the knob changes the offset or the position in the "Menu" > "Settings" > "Frontpanel" > "Knobs" dialog.

- Position indicates the vertical location in divisions.
- Offset moves the vertical center of the selected channel to the offset value.

If the selected waveform is a math or reference waveform, serial bus, or logic channel, the knob changes its vertical position.

The knob lights up in the color of the selected waveform. Turn clockwise to move up the waveform. To set the value to zero, press the knob.

The current value is shown in the input box in the upper right corner of the screen.

Remote command:
`CHANnel<m>:POSition` on page 1339
`CHANnel<m>:OFFSet` on page 1339

[Scale]
Sets the vertical scale in Volts per division. The vertical scale defines the displayed amplitude of the selected waveform.

The knob lights up in the color of the selected waveform. Turn clockwise to stretch the waveform. Doing so, the scale value V/div decreases. Press the knob to toggle between coarse and fine scale adjustment. For analog waveforms, the scale value is shown in the signal icon.

To get the maximum resolution of the waveform amplitude, make sure that the waveforms cover most of the diagram's height.

Remote command:
`CHANnel<m>:SCALe` on page 1338

3.2.3.5 Analysis keys

The keys in the Analysis functional block provide direct access to measurement and analyzing functions.

The rotary knob provides an alternative way to navigate in dialog boxes and to enter numeric data.

[Multiuse]
The multiuse knob has various functions. It lights up when a function is active on the knob.

- In numeric entry fields: turn to increase or decrease the value.
- To set cursor positions, histogram areas, and mask points in input boxes: press to toggle the parameter, turn clockwise to increase the value or turn counterclockwise to decrease it.
- To move zoom area, cursor line, or gate in diagrams: Turn to move the element that has the focus, and press to toggle the focus.
- In tables: press to activate the edit mode, turn clockwise to increase the value or turn counterclockwise to decrease it, and press to enter the value and move to the next cell.
- To change the intensity: Press the [Intensity] key and turn the knob.

[Intensity]
Adjusts the intensity of the waveforms on the screen, or the background transparency of dialog boxes, or the transparency of result boxes.

Press the [Intensity] key and turn the [Multiuse] knob. Press the [Intensity] key again to toggle between the three settings. The controlled parameter and its value are shown in the input box in the upper right corner of the screen.

[Cursor]
Starts a cursor measurement: displays vertical and horizontal cursor lines in the active diagram, and the results of the cursor measurement.

If you press the key while a cursor measurement is enabled, the "Cursors" dialog opens. The next press closes the dialog.

Cursors are markers which are placed at points of interest on a waveform. The instrument measures the cursor positions and delta values between parallel cursors.

[Measure]
Opens and closes the "Measurements" dialog and the subtab of the last used measurement. In the dialog, you can configure and start automatic measurements.

[History]
Opens and closes the history player. The history shows stored acquisitions that were acquired before the current one.

The key is illuminated as long as history acquisitions are displayed.

[HD]
Activates the high definition mode and opens the "HD Mode" dialog box.

[Autoset]
The instrument analyzes the enabled channel signals, and adjusts appropriate horizontal, vertical, and trigger settings to display stable waveforms.

[Preset]
Resets the instrument to a predefined state and starts the continuous acquisition. All measurements, mask tests, zoom, and most individual settings are deleted, and all channels except for channel 1 are disabled. You can define preset configurations and save them to a file. The [Preset] key can be configured to set either factory defaults or a user-defined preset configuration.

Camera
Performs the action that is assigned to the key in "Menu" > "Settings" > "Frontpanel" > "Hardkeys". By default, the key saves a screenshot of the waveform display.

See also: Chapter 4.5.1, "Hardkeys: function assignment", on page 94

[Touch Lock]
Locks the touchscreen to prevent unintended use. When the touchscreen is off, the key is illuminated. Press again to unlock the touchscreen.

[Clear]
Deletes all measurement results including long-term measurement and statistics, all waveforms, and the history.

App Cockpit
Opens the "Apps" dialog, where you can start an application, serial protocol or another analysis function.

3.3 Operating the instrument

There are three ways to operate the R&S RTO6.

Manual operation

Use the touchscreen, keys and rotary knobs, or an optional mouse and/or keyboard. The principles of manual operation are explained in this section.

Remote control

Create programs to automatize repeating settings, tests, and measurements. The instrument is connected to a computer that runs the program.

This way of operation is described in: Chapter 24, "Remote control commands", on page 1258

Remote operation

The remote desktop connection of Windows can be used for instrument control and file transfer. Even on computers with non-Windows operating systems, a remote desktop connection is possible using RDP applications.

For details, refer to the user manual, chapter "Remote Desktop Connection".

Remote monitoring and control of the instrument from a connected computer is also possible with a standard web browser and a LAN connection.

For details, refer to the user manual, chapter "Web Control".

Alternatively, you can use Virtual Network Computing (VNC), which requires installation of the VNC server on the R&S RTO6. Installation and configuration are described in the application note "Remote Monitoring and Control of the R&S RTO6 with a Web Browser", available on the Rohde & Schwarz internet site.

3.3.1 Means of manual interaction

The R&S RTO6 provides the following means of manual interaction, which you can use alternatively or complementary:

- Touchscreen:
  Using the touchscreen allows for direct interaction with the instrument. Use your finger to place waveforms on the screen, mark areas for zoom, set parameters in dialogs, enter data, and much more. The control elements and actions on the screen are based on common concepts, and you easily become familiar with the user interface.

Tapping the screen works like clicking mouse buttons:

– Tap = click: Selects a waveform or parameter, or provokes an action.
– Double-tap = double-click: Has the same effect as touch and hold, it opens the on-screen keyboard or keypad, or a specific editor if available.

Use gestures to scale the waveform:

– Spread or pinch two fingers horizontally to change the time scale.
– Spread or pinch two fingers vertically to change the vertical scale of the active waveform.

- Function keys and rotary knobs:
  The front panel provides frequently used functions and controls to operate the instrument. Use knobs to set levels and scales, and keys to initiate actions and to open dialogs.
- Optional mouse and/or keyboard
  These devices work in the normal manner.

3.3.2 Touchscreen display

3.3.2.1 Information on the display

The touchscreen of the instrument shows the waveforms and measurement results, and also information and everything that you need to control the instrument. All waveform-related display elements are shown in Figure 3-3. An overview of control elements - like dialog box, toolbar - is given in Chapter 3.3.2.2, "Control elements on the touchscreen", on page 46.

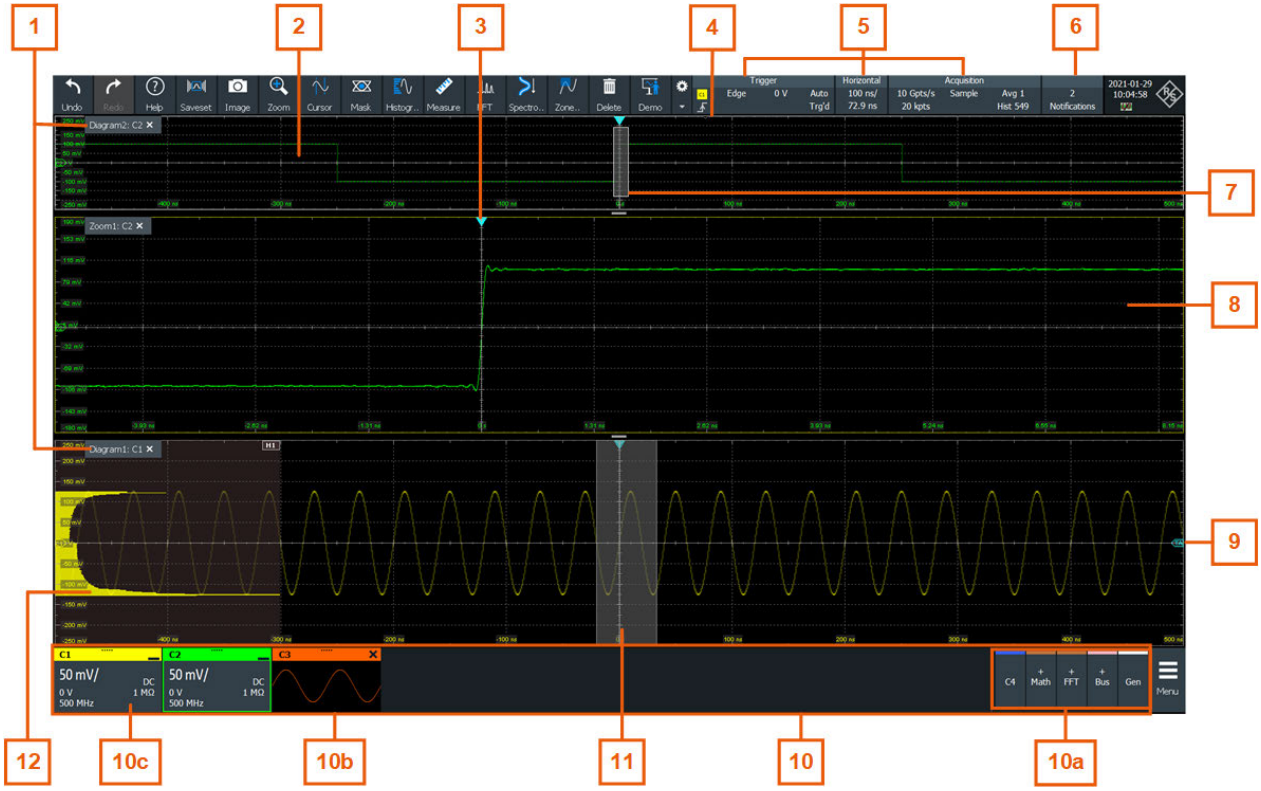


Figure 3-3: Display information

1 = Diagram
2 = Grid
3 = Trigger position
4 = Reference point (distance from trigger position to reference point = horizontal position)
5 = Trigger, horizontal and acquisition label
6 = Notifications
7 = Zoom area
8 = Zoom diagram
9 = Trigger level
10 = Signal bar with inactive waveforms (10a), minimized live waveform (10b) and signal icon with vertical settings (10c)
11 = Histogram area
12 = Histogram

Diagram

A diagram shows one or more waveforms: channel, reference, and math waveforms together with histograms, masks etc. Zoom details, XY-waveforms, spectra and other special waveforms are shown in separate diagrams.

By default, the diagram name contains the diagram number and the short names of the waveforms shown inside. To change the diagram name, touch and hold the tab name. The on-screen keyboard opens to enter the new name. Names must be unique.

To arrange the diagrams on the screen, the Rohde & Schwarz SmartGrid function helps you to find the target place simply and quickly. For details, see Chapter 3.3.5, "Rohde & Schwarz SmartGrid", on page 50. You can also adjust the diagram size by dragging the diagram border.

Grid

The grid shows the vertical and horizontal divisions. The division lines are labeled with the correspondent values. The grid labels have the color of the waveform to which they belong. If several waveforms are shown in one diagram, the grid has the color of the selected waveform.

Trigger position and trigger level

The blue markers show the horizontal position of the trigger and the vertical trigger level. You can touch and move the trigger markers in the diagram to set the positions. The trigger point is the zero point of the diagram.

The trigger position can be moved outside the diagram. A red trigger position marker indicates that the trigger position is not visible.

Trigger, Horizontal, Acquisition

The "Trigger", "Horizontal" and "Acquisition" labels show the main timebase and trigger settings. If you tap a label, the relevant dialog box opens.

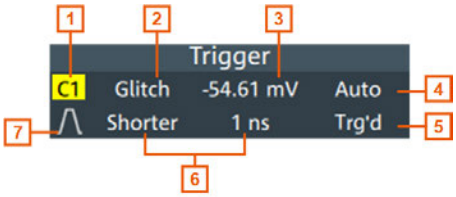


Figure 3-4: Trigger label on the toolbar

1 = Trigger source
2 = Trigger type
3 = Trigger level
4 = Trigger mode
5 = Trigger state
6 = Trigger slope

Figure 3-5: Horizontal label on the toolbar

1 = Time scale
2 = Horizontal position

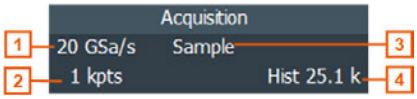


Figure 3-6: Acquisition label on the toolbar

1 = Sample rate
2 = Record length
3 = Decimation
4 = Number of acquired waveforms

Reference point

The reference point marks the rescaling center. If you modify the time scale, the reference point remains fixed on the screen, and the scale is stretched or compressed to both sides of the reference point.

You can define the position of the reference point ("Menu" > "Horizontal" > "Reference point"), and also its time distance from the trigger point of the diagram (Position knob).

Notification

The "Notification" button on the toolbar points to the status messages of the instrument. To open the message box, tap the button. See also: Chapter 3.3.10, "Instrument information and notifications", on page 63.

Zoom diagram and zoom area

Zoomed waveforms are shown in separate zoom diagrams, in addition to the waveform diagrams. On the original waveform diagram, a rectangle indicates the zoomed section of the waveform - this is the zoom area. You can modify the zoom area by dragging the rectangle as a whole, and by dragging its edges. To toggle between these modes, tap the zoom area. You can also set exact positions.

The frames of the zoom area and of the associated zoom diagram have the same color, different zooms are marked with different colors. So it is easy to assign zoom area and zoom diagram.

As for waveform diagrams, you can change the name of the zoom diagram. A zoom in a zoom and coupled zooms are also possible.

For details, see Chapter 7.1, "Zoom", on page 239.

Signal bar

The signal bar is the control center for all waveforms. All enabled waveforms are shown on the left side of the signal bar. On the right side of the signal bar, you see the signal activators of inactive waveforms. Tap a signal activator to enable the waveform.

Each waveform is represented by a signal icon. If the waveform is shown in a diagram, the signal icon displays its main vertical and acquisition settings. If you tap a signal icon, the dialog with vertical settings for this waveform opens. If you tap the "Minimize" icon on the signal icon, the waveform switches from the diagram area to the signal icon: the icon shows the real-time signal preview. See Chapter 3.3.4, "Working with waveforms", on page 48 for a detailed description.

In Figure 3-3, the signal icons C1 and C2 show the main settings, and the waveforms are displayed in diagrams. Other waveforms are minimized and shown as signal preview.

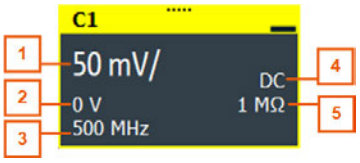


Figure 3-7: Signal label

1 = Vertical scale
2 = Offset
3 = Bandwidth
4 = Coupling
5 = Termination

If the signal bar contains many signal icons, and not all icons are visible, touch one of the icons and scroll left or right until the required icon appears.

You can also switch off the signal bar: "Menu" > "Settings" > "Appearance" > "Diagram" > "Show signal bar".

Histogram and histogram area

A histogram shows the frequency of occurrence of voltage or time values in a bar chart directly in the diagram. The rectangular histogram area indicates the part of the waveform that is considered in the histogram. The vertical histogram counts the voltage values, and the horizontal histogram counts time values. You can switch between vertical and horizontal mode, and modify the histogram area by dragging the rectangle as a whole, by dragging its edges, or by setting exact positions.

3.3.2.2 Control elements on the touchscreen

The touchscreen provides everything you need to control the instrument, to analyze waveforms, and to get measurement results. Figure 3-8 shows the control elements at a glance.

Figure 3-8: Control elements on the touchscreen

1 = Toolbar
2 = Tab in a dialog box
3 = Dialog box
4 = Result table (docked)
5 = Signal bar
6 = Menu
Not shown = Input box

Toolbar (1)
The icons on the toolbar provide quick and easy access to the most important functionality. For a detailed description, refer to Chapter 3.3.6, "Toolbar", on page 52.

If you adjust the settings of an analyzing function, e.g., cursor measurement, the overlay menu is shown instead of the icons. The overlay menu provides the most important settings of the current action. If you need more settings, "Advanced Setup" opens the corresponding dialog box.

Dialog box (2, 3)
The tabs of the dialog boxes contain all task-oriented settings and operations, and black buttons for calling related tabs. The usage of dialog boxes is described in Chapter 3.3.8, "Using dialogs", on page 60.

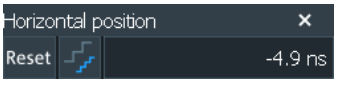
Result table (4)
If you perform cursor or automatic measurements, mask testing, or a search, the result table shows the results of the action. The position of the result table is adjustable. It can be docked (default for measurements), floating, minimized to a result icon on the signal bar, or displayed in a separate diagram on the screen.

See also: Chapter 3.3.7, "Displaying results", on page 58.

Signal bar (5)
The signal bar summarizes all waveforms as described in "Signal bar" on page 45.

Menu (6)
The menu provides access to the complete functionality of the R&S RTO6.

Input box
The input box appears if you adjust a value using one of the rotary knobs, or if you drag an element on the screen, for example, a cursor line. The input box shows the current value of the modified parameter. You can enter the exact numerical value, change the step size, and - if available - autoset the value directly in the input box. The box title shows the name of the currently adjusted parameter.



See also: Chapter 3.3.9, "Entering data", on page 61.

3.3.3 Applications

All available applications for special tasks are provided in the "Apps" dialog.

► To open the "Apps" dialog:

- Open "Menu" > "Apps".
- Press the app key on the frontpanel.

3.3.4 Working with waveforms

The R&S RTO6 can create and display many waveform types. The most importand are:

- Channel waveforms:
  Up to three waveforms per input channel can be shown. For a four-channel instrument, 12 channel waveforms are available.
- Reference waveforms:
  Four waveforms can be used as reference for comparison and analysis.
- Math waveforms:
  Eight mathematic waveforms can be created with mathematic operations performed on channel, reference, and other math waveforms.
- Zoom waveforms:
  Show the details of waveforms.
- XY-waveforms:
  Four XY-waveforms can be created. Each XY-waveform is built from the voltage values of two source waveforms.

- Digital waveforms:
  The Mixed Signal Option R&S RTO6-B1 provides 16 digital channels grouped in two logic probes (pods) with 8 channels each.

Waveform handling

The R&S RTO6 can show and analyze many waveforms. To handle this multitude while keeping track of it, the R&S RTO6 provides intelligent support:

- The color system helps to distinguish the waveforms. The color of the vertical rotary knobs indicates the signal that is focused (selected). The color of each waveform can be changed, the color of its signal icon and of the illuminated keys is adjusted to the new color. Alternatively, a color table can be assigned to a waveform.
  Settings: "Menu" > "Settings" > "Appearance" > "Colors" tab.
- Waveforms can be minimized to signal icons showing a real-time signal preview. Thus, more space in the diagram area is available without switching off waveforms.
- Diagrams are displayed on tabs – you can arrange them side by side or one above the other. To change the diagram name, double-tap the tab name.
- The Rohde & Schwarz SmartGrid function helps to arrange the diagrams.

Waveform states

Depending on its place on the screen and the effect of settings, a waveform has one of the following states:

- Off
- Active: The waveform is shown in a diagram.
- Selected:
  One of the active waveforms that has the focus. In each diagram, one of the assigned waveforms is selected – it appears "on top" in the diagram, and the grid labels have the color of the selected waveform. Some of the toolbar functions, like cursor and histogram measurements, are performed on the selected waveform. All waveform-specific settings are applied to the selected waveform of the selected diagram.
  The vertical [Position] and the [Scale] knobs are illuminated in the color of the selected waveform.
  In Figure 3-3, C2 is the selected waveform: The frame of the diagram and the signal icon are highlighted.
- Minimized:
  The waveform is shown as real-time signal preview in its signal icon.

To switch a waveform on

A channel waveform is activated as soon as you connect the probe. You can switch it on and off according to your needs.

► Choose one of the following ways:

- Press the channel key.
- Tap the signal activator on the right side of the signal bar.
- In the "Vertical" dialog box, select the channel and tap "Channel" > "On".

The waveform is now active, selected, and is shown in the diagram.

Remote command: `CHANnel<m>:STATe` on page 1337

To select a waveform

► Choose one of the following ways:

- Tap the waveform in the waveform diagram.
- To select a channel, reference, or math waveform, press the corresponding key.
- Tap the signal icon.

Note: Zoom waveforms in zoom diagrams cannot be selected.

To minimize a waveform

► Choose on of the following ways:

- Tap the "Minimize" icon in the upper right corner of the waveform's signal label in the signal bar.
- Drag the waveform from the diagram to the signal bar.

The waveform disappears from the diagram and the signal preview is shown in the signal icon.

Remote command: `LAYout:SIGNal:UNASsign` on page 1306

► To set the waveform back to its previous diagram immediately, use "Undo".

To arrange a waveform using the SmartGrid

See Chapter 3.3.5, "Rohde & Schwarz SmartGrid", on page 50.

To switch off a waveform

► Do one of the following:

- To switch off a signal preview, tap the "Close" icon in the upper right corner of the minimized signal view.
- Disable "Channel" in the "Vertical" > "Setup" tab.
- Tap the "Delete" icon (Recycle bin) in the toolbar, and then the waveform.
  If several waveforms overlap or lie close together, the upper (selected) waveform is switched off.

Remote command: `CHANnel<m>:STATe` on page 1337

3.3.5 Rohde & Schwarz SmartGrid

The Rohde & Schwarz SmartGrid helps to create and arrange the diagrams on the screen with drag&drop. The diagram layout depends on the position where you drop the signal icon, in relation to an existing diagram.

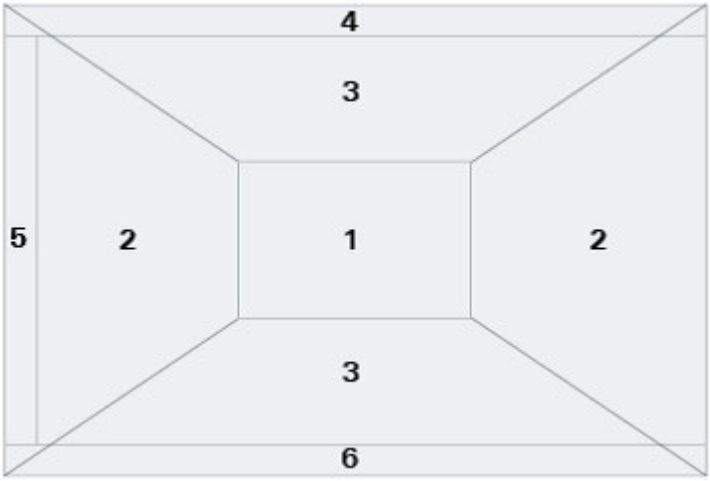


Figure 3-9: SmartGrid positions

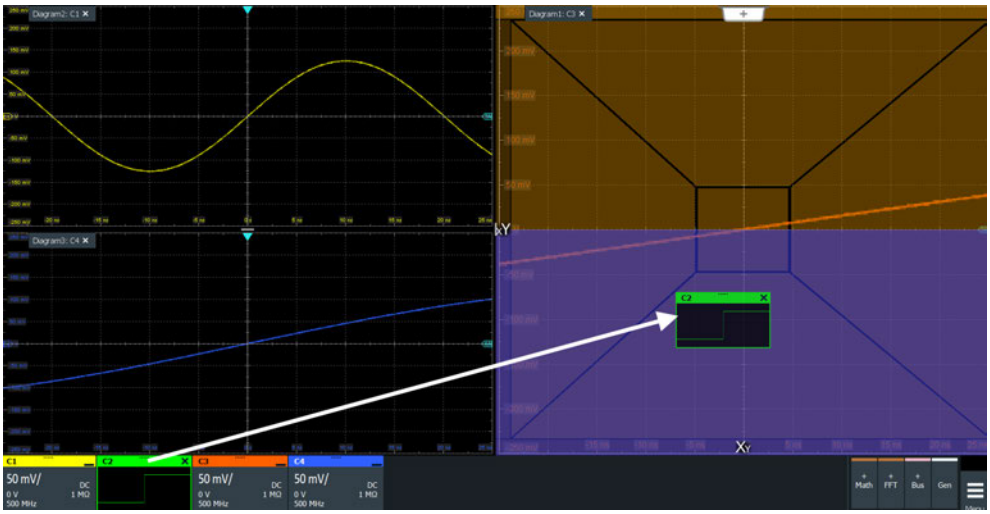
1 = In the existing diagram, overlay of signal
2 = New diagram on the left or right
3 = New diagram above or below
4 = New diagram on top of the existing diagram
5 = XY-diagram
6 = YX-diagram

The diagram configuration is deleted when you use [Preset] and `*RST`.

To arrange a waveform using the SmartGrid

You can arrange waveforms in one of the existing diagrams, or in a new diagram.

1. Drag the signal icon to the diagram area, and move it around.
   The Rohde & Schwarz SmartGrid appears and a blue area shows where the waveform will be placed.
2. Drop the waveform in the target area.
   The waveform appears in an existing or in a new diagram, and it is selected for further actions.

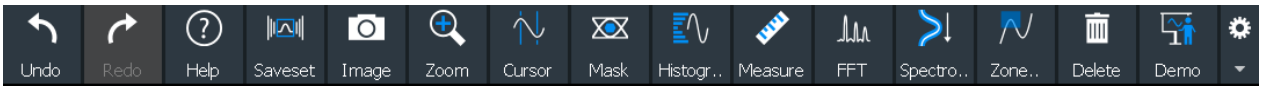


3. To change the size of a diagram, tap the handle between two diagram frames and drag it to the required position.

Remote commands: see Chapter 24.7.2.4, "Smartgrid", on page 1301.

3.3.6 Toolbar

The toolbar provides direct access to important control and measurement functions.



By default, the toolbar shows the most frequently used functions. You can configure the content of the toolbar, see Chapter 3.3.6.2, "Configuring the toolbar", on page 53.

3.3.6.1 Using the toolbar

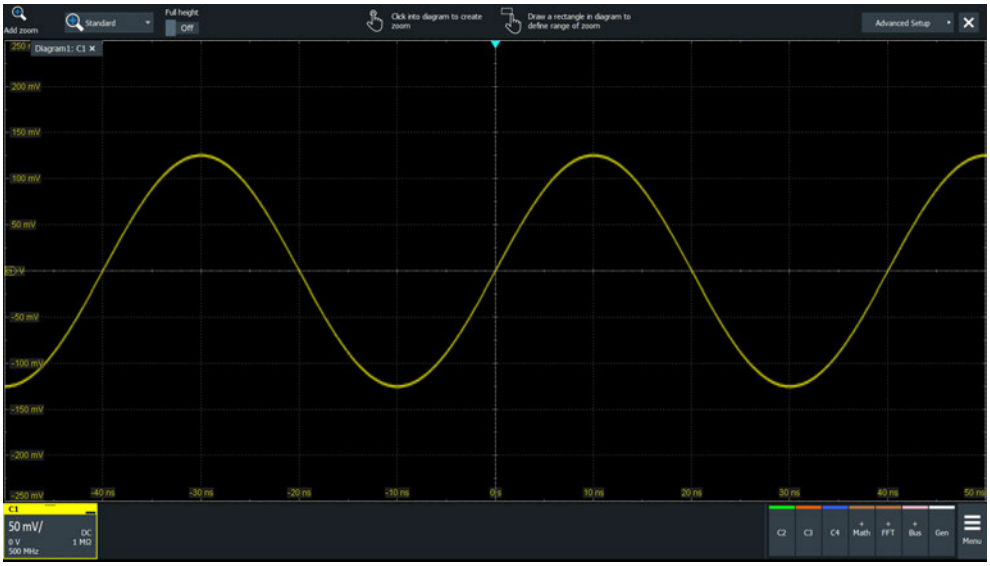
Using the toolbar is easy and straightforward.

Some of the toolbar functions are one-click actions. These actions are performed immediately when you tap the icon.

Other toolbar functions are analyzing functions. These functions are interactive actions.

To use analyzing functions (interactive actions)

1. If several waveforms are shown in the diagram, select the waveform that you want to analyze.
   See: "To select a waveform" on page 50
2. Tap the icon of the function in the toolbar.
   If there are many functions on the toolbar, and not all icons are visible, scroll the toolbar using the arrow on the right end of the toolbar.
3. Check and adjust the settings in the overlay menu.



4. To define the analyzed area, do one of the following:
   - Tap the required diagram.
   - Drag a rectangle on the diagram.

3.3.6.2 Configuring the toolbar

You can configure the content of the toolbar so that only the required functions are displayed. The toolbar configuration is part of the user preferences. It is retained when you switch off and on the instrument, and you can save it in the user preferences and user-defined preset.

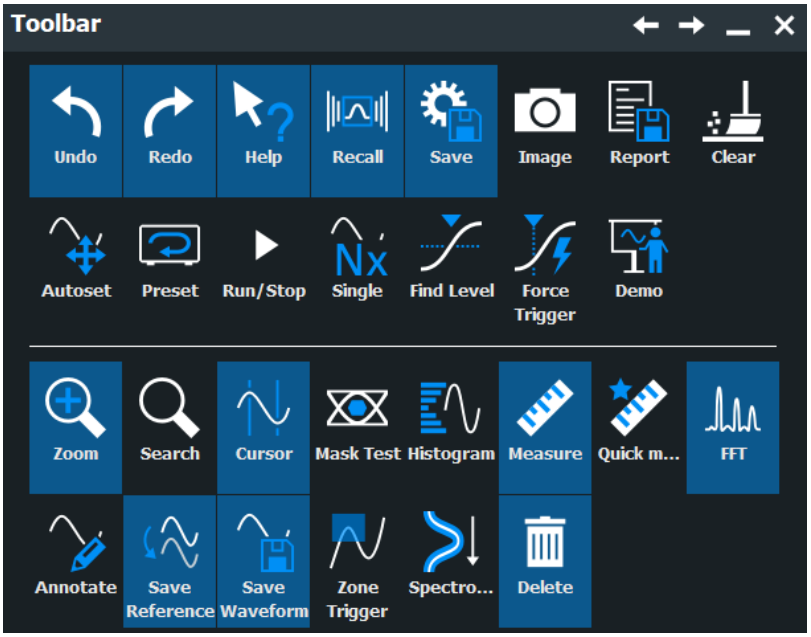
1. To open the toolbar configuration, tap the icon in the toolbar:

2. Select the required toolbar functions:
   - Disable the functions that you do not need.
   - Enable the functions that you want to add to the toolbar.
3. Close the dialog.

A detailed description of the toolbar functions is given in Chapter 3.3.6.3, "Toolbar functions", on page 53.

3.3.6.3 Toolbar functions

This section describes all toolbar functions in detail.



One-click actions	Interactive actions
Undo	Zoom
Redo	Search
Help	Cursor
Recall	Mask Test
Save	Histogram
Image	Measure
Report	Quick meas
Clear	FFT
Autoset and Preset	Annotate
Run / Stop and Single	Save Reference
Find Level	Save Waveform
Force Trigger	Zone trigger
Demo	Spectrogram (option R&S RTO6-K37)

You can configure the content of the toolbar, see Chapter 3.3.6.2, "Configuring the toolbar", on page 53.

The following list describes at first the default toolbar functions and then the additional functions.

Undo
Undoes the last setting actions step by step. Some actions cannot be revoked: locking the touchscreen with [Touch Lock], and saving data. The undo stack is deleted during the following actions: Reloading settings from file, reference waveform actions (save, load and preset with active reference waveform), and some actions that require an internal re-alignment to ensure correct measurements.

Redo
Recovers the undo steps in reverse order.

Help
Enables the tooltip display. A short description appears when you tap a parameter in a dialog. To open the corresponding help topic, tap the "Show Help" button in the lower right corner of the tooltip. See also: Chapter 3.3.11, "Getting help", on page 64.

Recall
Opens a window to select and load instrument settings that were previously stored in a saveset. A graphical preview helps you to find the required settings.

Save
Saves the current instrument settings in a saveset.

You can reload the saveset using the "Recall" toolbar icon, or using "Menu" > "Save/Recall" > "Recall" tab > "Saveset with preview".

The filename is created according to the autonaming pattern, defined in "Menu" > "Settings" > "Save/Recall" > "Autonaming" tab.

Image
Saves a screenshot of the current display using the settings defined in "Menu" >"Save/Recall" > "Save" tab > "Screenshot".

Report
Creates a report of the current measurement settings and results using the settings defined in "Menu" > "Save/Recall" > "Save" tab > "Report".

Clear
Deletes all measurement results including long term measurement and statistics, all waveforms, and the history.

Remote command:
`DISPlay:CLR` on page 1321

Autoset and Preset
Performs an autoset, or a preset to a default state. The icons have the same functionality as the corresponding keys on the front panel. They are useful when you operate the instrument remotely.

Run / Stop and Single
Starts and stops the continuous acquisition, or starts a defined number of acquisition cycles. The icons have the same functionality as the corresponding keys on the front panel. They are useful when you control the instrument remotely.

Find Level
Analyses the signal and sets the trigger level to the middle of the signal peaks.

Force Trigger
Starts an immediate single acquisition. If the acquisition is running in normal mode and no valid trigger occurs, use "Force Trigger" to confirm that a signal is available. Then you can use the displayed waveform to determine how to trigger on it.

Demo
Opens the "Demo" dialog, where you can find examples of operating the R&S RTO6.

Zoom
Creates a zoom. Tapping the icon opens the overlay menu, where you select one of the four zoom types: standard zoom, hardware zoom, coupled zoom, fingertip zoom.

Standard zoom ← Zoom
Displays a magnified section of the diagram in an additional zoom diagram. It is a display zoom, instrument settings are not changed.

Touch and hold the zoom area to open the "Zoom" dialog box.

Remote command:
`LAYout:ZOOM:ADD` on page 1438

Hardware zoom ← Zoom
Changes the instrument settings - horizontal and vertical scales as well as trigger level and offset - to display a part of the diagram in greater detail.

Coupled zoom ← Zoom
Creates a coupled zoom area and its related zoom diagram. If you change the size of one zoom area, the size of all coupled zoom areas is changed as well.

Remote command:
`LAYout:ZOOM:ADDCoupled` on page 1438

Fingertip zoom ← Zoom
Magnifies the waveforms around your fingertip.

Tap the icon and put your finger on the waveform. The touched part of the waveform is displayed in a magnifier. Drag your finger on the screen to move the magnifier. You can change the zoom factor using the [Multiuse] knob.

Search
Performs a search according to the settings in the "Search Setup" dialog box. Tap the icon and then tap the diagram with the waveform to be searched, or drag a rectangle to define a search gate. The search is performed on the selected waveform.

Cursor
Performs a cursor measurement. Tapping the icon opens the overlay menu, where you select the cursor type and source.

Tap the diagram where you want to set the cursors, or draw a rectangle in the diagram to position the cursor lines. The resulting cursor lines measure the selected waveform, and the results are shown.

Mask Test
Starts the on-screen mask definition and the testing against the defined mask.

Tap the icon and then tap the points that build the mask. To finish the shape definition, tap "Finish segment" in the overlay menu. To create a rectangular mask, draw a rectangle on the screen. You can add more segments to the test, and add another test. You can also move the mask segments on the screen.

To configure the mask test settings, tap the icon in the "Mask" result box.

Histogram
Creates a histogram. Tapping the icon opens the overlay menu, where you select the histogram type and source.

Drag a rectangle on the diagram to define the histogram area. The histogram for the selected waveform appears.

To change settings, touch and hold the histogram area, and the histogram overlay menu opens.

Measure
Performs automatic measurements. Tapping the icon opens the overlay menu, where you select the category, the measurement types and measured waveform. Tap the diagram with the waveform to be measured. To define a measurement gate, draw a rectangle on the screen.

You can run up to 8 automatic measurement groups in parallel. The "Measure" icon starts the measurement groups one after the other. A measurement group consists of minimum one measurement, and can consist of many measurements of the same category.

To modify the measurement, double-tap one of the result values.

Quick meas
Performs a set of measurements on the selected waveform or on the selected gate. You can configure up to 8 measurements to be included in quick measurement.

Tap the icon and then tap the waveform to be measured or draw a rectangle to define the gate.

FFT
Transforms a waveform to the frequency spectrum by fast Fourier transform (FFT). The FFT trace is shown in a new diagram.

Tap the icon and adjust the settings in the overlay menu. Select the source of the spectrum in the overlay menu. The FFT diagram is created from the selected waveform.

To adjust FFT settings, double-tap the FFT diagram.

Annotate
Defines a waveform label that names or explains the waveform. Tap the icon and then tap the waveform to be labeled. Enter the label text using the on-screen keyboard. The text is shown in the same color as the waveform. If you tap the display background, the label is assigned to the focused waveform. You can drag the label to another position.

Save Reference
Copies the selected source waveform with all its settings to the reference waveform. If the acquisition is running, the reference waveform is a snapshot. You can configure up to four reference waveforms.

Tap the icon, and select the required reference waveform (R1 to R4) in the overlay menu. Tap the waveform to be copied.

Save Waveform
Exports the waveform data to file using the settings defined in "Menu" > "Save/Recall" > "Save" tab > "Waveform" > "Setup" tab.

The filename is created according to the autonaming pattern.

Tap the icon and then tap the waveform to be exported. If you tap the display background, the focused waveform is exported, or a multichannel export is performed if configured.

Zone trigger
Defines a zone trigger, which combines the trigger condition with the intersection or non-intersection of one or more zones or masks.

Tap the icon and then tap the corner points of the zone on the screen. Tap "Finish zone". You can add more zones to the trigger condition.

Spectrogram (option R&S RTO6-K37)
Starts an FFT and the spectrogram. The FFT trace and the spectrogram are shown in separate diagrams.

Tap the icon. Select the source in the sidebar, or tap diagram with the waveform to be transformed. The diagrams are created from the selected waveform.

Delete
Removes waveforms, measurements, zoom, histograms, mask segments and other elements from the display.

Tap the icon and then tap the element to be deleted, or the waveform to be switched off.

3.3.7 Displaying results

The results of measurements, mask tests, searches, protocol decoding and others are displayed immediately. There are several places to display the results:

- In a table that is docked below the diagram (only for measurement and decode results)
- In a floating result box in front of the diagrams, which you can move on the display

- In a minimized view (result icon) on the signal bar
- In a separate tab that can be arranged similar to a diagram

The default position and the font size can be adjusted.

To define the default position of measurement results

For results of automatic, cursor and protocol measurements and for decode results, the docked table below the diagram is the initial default position.

To change the default position of the results:

► Open the "Menu" > "Settings" > "Appearance" dialog.

- Automatic measurements: select "Measurement" tab > "Result position".
- Cursor measurements: select "Cursor" tab > "Result position".
- Protocols: select "Protocol" tab > "Result position".

Mask test and search results are always shown in a floating result box, which you can drag to a convenient position using the SmartGrid.

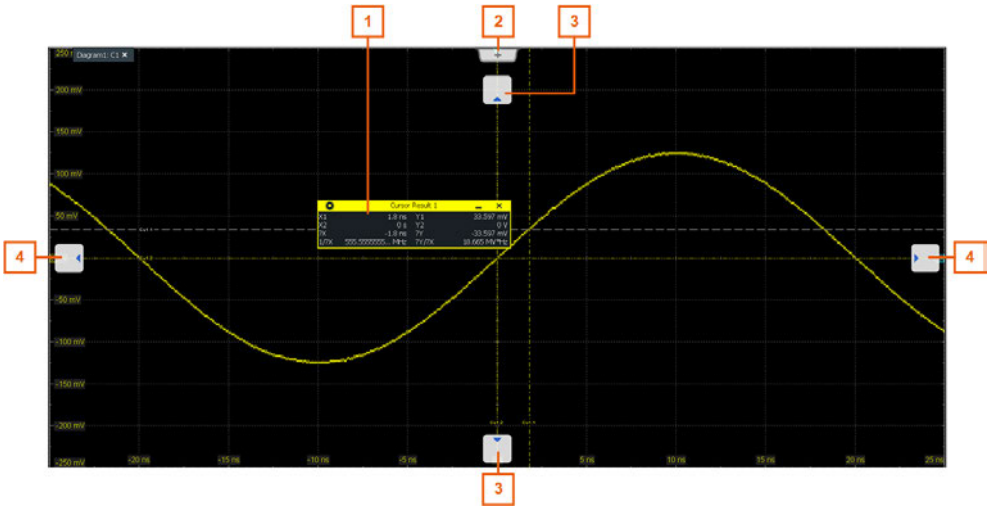
To arrange the results on the display

1. If the results are listed in a docked table, touch and hold a result line and drag it.

   The results are now shown in a floating result box.

   Note: Once you undock the measurement results, you cannot move the result box back to the docked position.

2. Touch and hold the title of the result box and drag the box on the screen. The SmartGrid indicates where the result box will be placed.
   - If you drop the box on one of the buttons, the results are shown in a separate tab besides, above, or below the diagram.
   - If you drop the box on the signal bar, a result icon is created.
   - If you drop the box somewhere else, the floating result box is placed there.



1 = Floating result box
2 = New tab
3 = Table in a tab above or below
4 = Table in a tab on the left or right

To open the corresponding settings

1. Double-tap one of the result values.
   If a result box is shown, you can also tap the ⚙ icon.

   The overlay menu appears and provides the most important settings.

2. In the overlay menu, tap "Advanced Setup".

   The corresponding dialog opens.

To adjust the font size of results

1. Open the "Menu" > "Settings" > "Appearance" dialog.
2. Select the "Dialogs" tab.
3. Set the "Result dialog" > "Font size".

3.3.8 Using dialogs

Dialogs provide groups of related settings and allow to make selections and enter data in an organized way.

To open a dialog

► Perform one of the following actions:

- Open the "Menu", and select the menu entry.
- Press the function key on the front panel.
- If the signal bar is open, tap "Advanced Setup".
- To open the "Vertical" dialog of a waveform, tap the signal icon.
- Tap the "Horizontal", "Acquistion" or "Trigger" label to open the corresponding dialog.

For direct access to important control and measurement functions, use the toolbar, see Chapter 3.3.6, "Toolbar", on page 52.

Each dialog has four icons in the upper right corner:

←	Go back: opens the previously opened dialog box.
→	Go forward: opens the next dialog box.

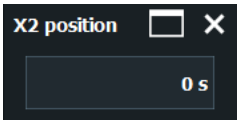
	Minimizes the dialog box to a small box that only contains the last selected function.
	Closes the dialog box.

To minimize a dialog

If you want to change only one setting during analysis, and you need to change it often, you can display a small box that only contains the required setting.

1. Open the dialog.
2. Tap the setting that you need to modify.
3. Tap the "Minimize" icon in the upper right corner of the dialog box.

   The dialog box turns into a small box that contains only the setting, that was last selected.

4. To restore the complete dialog box, tap the "Maximize" icon in the small box.

3.3.9 Entering data

To set parameter values and enter other data, you use the various knobs and the on-screen keypad or keyboard.

Using scale, position and level knobs

The instrument has dedicated rotary knobs to set vertical and horizontal positions and scales, and the trigger level. When you turn a knob, the input box opens in the upper right corner of the screen, showing the parameter name and current value.

1. Turn the knob to change the value.
2. Press the knob:
   - [Scale]: toggles the increment.
   - [Position]: sets to zero.
   - [Level]: sets the trigger level to 50% of the signal.
3. The input box also has settings:
   - Tap the "Steps" icon to toggle the increment.
   - Tap the "Reset" icon to set the parameter to the autoset value (if available).

To enter values with the on-screen keypad

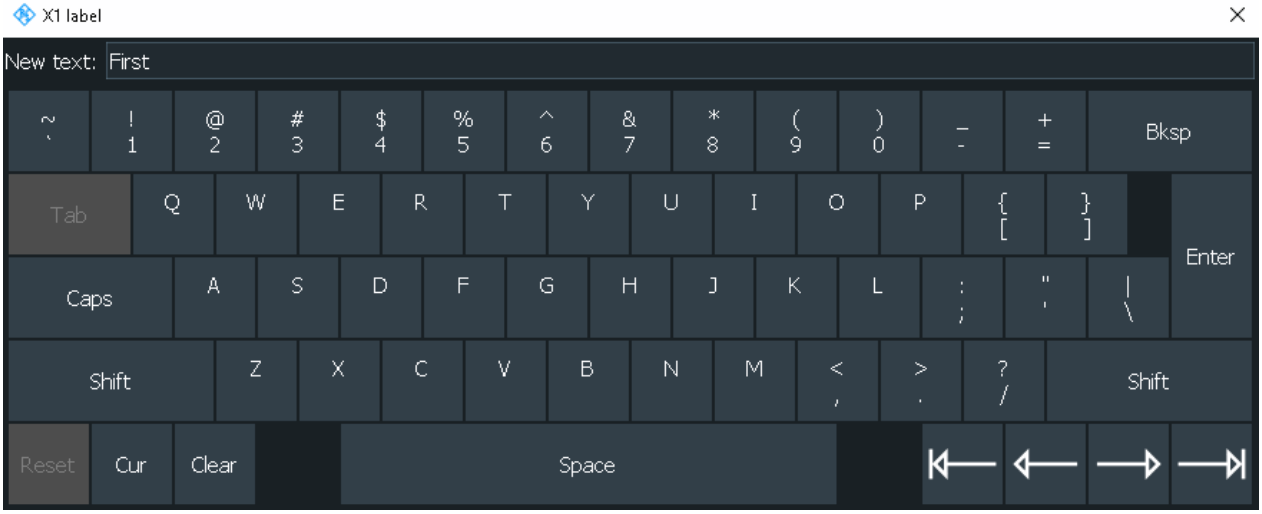
To enter numeric values and units in dialogs, the touchscreen provides an on-screen keypad. For text input, the on-screen keyboard with English key layout is used.

1. Double-tap the entry field.
   The on-screen keypad opens.

2. Enter the numeric value using one of the following methods:
   - To use the default value, tap "Reset" (if available).
   - To use the minimum or maximum value, tap "Min" or "Max", respectively.
   - To increase the displayed value in fixed steps, tap "Inc".
     To decrease the value in fixed steps, tap "Dec".
     To toggle between small steps and large steps, tap the "Steps" icon.
   - To get the value that was used before the keypad opened, tap "Cur".
   - To enter a user-defined value, tap the numbers and complete the entry by tapping the unit button.
     – The arrow buttons move the cursor left or right.
     – "Bksp" deletes the last character before the cursor.
     – "±" changes the sign of the value.

To enter data with the on-screen keyboard

1. Double-tap the entry field to open the on-screen keyboard.
   If available, you can also tap the keyboard icon on the right of the entry field.

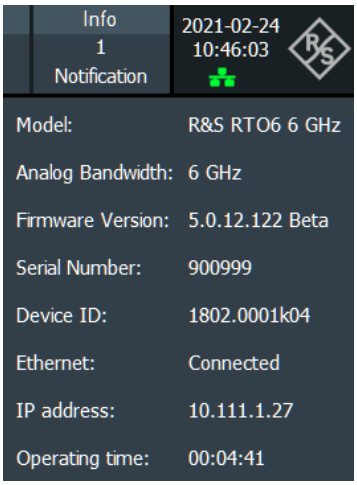


2. Enter the text as you would do on a normal keyboard.
   - To enter a series of capital characters, tap "Caps".
     To enter one capital character, tap "Shift".
   - To get the value that was used before the keyboard opened, tap "Cur".
   - The arrow buttons move the cursor left or right.
   - "Bksp" deletes the last character before the cursor.
3. Tap "Enter" to complete the entry.

3.3.10 Instrument information and notifications

In the upper right corner of the screen, you see the Rohde & Schwarz logo, date and time, the symbolic information on LAN connection and the notifications status.

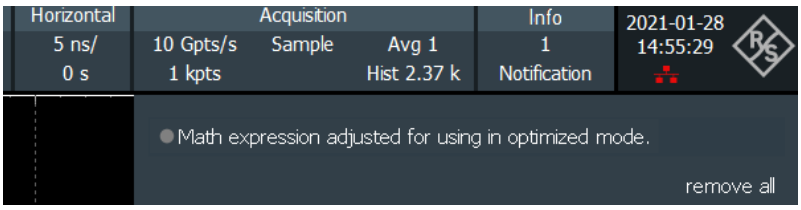
► To see the instrument information, tap the Rohde & Schwarz logo.



Notifications are status messages, information on mismatching settings and similar information. They are displayed for a few seconds and saved.

The color of the dot before the text indicates the severity: gray for information, orange for warnings, and red for errors.

► To read the notifications, tap "Notification".
You can also delete the list of notifications.



1. To hide the date and time, open "Menu" > "Settings" > "Appearance" > "Diagram" tab.
2. Disable "Show date / time".

3.3.11 Getting help

If you need information on the instrument's functionality, you can use the following sources:

- Tooltips give a short description of the parameter.
- The context help provides functional description on a setting, and the corresponding remote command.
- The general help opens the help window with a table of contents, where you can browse and search for information.

3.3.11.1 Displaying help

To display tooltips and context help

1. Enable the "Help" icon on the toolbar.

2. Tap the parameter for which you need information.

   The tooltip opens.

3. To open the corresponding help topic, tap the "Show Help" button in the lower right corner of the tooltip.

   The "Help" window opens and displays the comprehensive description and the corresponding remote command. You can browse the help for further information.

   Note: The "Help" icon disables automatically when you tap a parameter. To show another tooltip, tap the "Help" icon again.

To open general help

1. Open the "Menu".
2. Tap the "Help" icon.

   The help window opens with the "Contents" page, where you can select the topics.

3.3.11.2 Using the Help window

The Help window contains several tabs:

- "View" - shows the selected help topic
- "Contents" - contains a table of help contents
- "Index" - contains index entries to search for help topics
- "Search" - provides text search

The Help toolbar provides some buttons:

- To browse the topics in the order of the table of contents: up arrow = previous topic, down arrow = next topic
- To browse the topics visited before: left arrow = back, right arrow = forward
- To increase or decrease the font: magnifier with + or - sign, respectively

To navigate the Help, use the touchscreen.

To search for a topic in the index

The index is sorted alphabetically. You can browse the list, or search for entries in the list.

1. Switch to the "Index" tab.
2. Select the "Keyboard" icon besides the entry field.
3. Enter the first characters of the keyword you are interested in.

   The entries containing these characters are displayed.
4. Double-tap the suitable index entry.

   The "View" tab with the corresponding help topic is displayed.

To search topics for a text string

1. Switch to the "Search" tab.
2. Select the "Keyboard" icon besides the entry field.
3. Enter the string you want to find.
   If you enter several strings with blanks between, topics containing all words are found (same as AND operator).

For advanced search, consider the following:

- To find a defined string of several words, enclose it in quotation marks. For example, a search for *"trigger qualification"* finds all topics with exactly *"trigger qualification"*. A search for *trigger qualification* finds all topics that contain the words *trigger* and *qualification*.
- Use "Match whole word" and "Match case" to refine the search.
- Use operators AND, OR, and NOT.

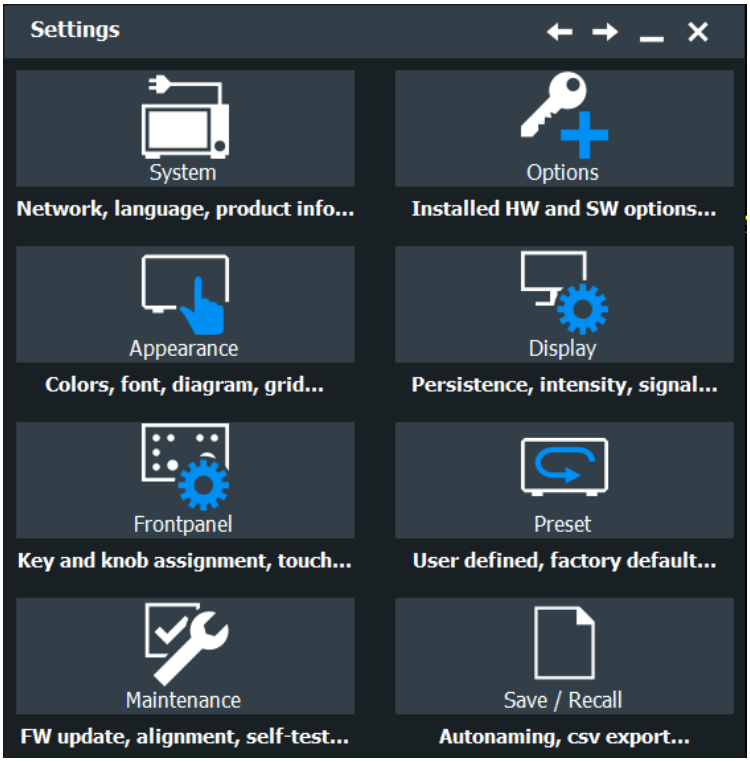
To close the Help window

► Select the "Close" icon in the upper right corner of the help window.

4 Instrument setup

Access: "Settings" menu > "Settings" dialog.

In the "Settings" dialog, you can adapt various instrument settings to your requirements, such as language, display appearance, and assign functions to some keys.



The following settings and procedures are described in the current chapter:

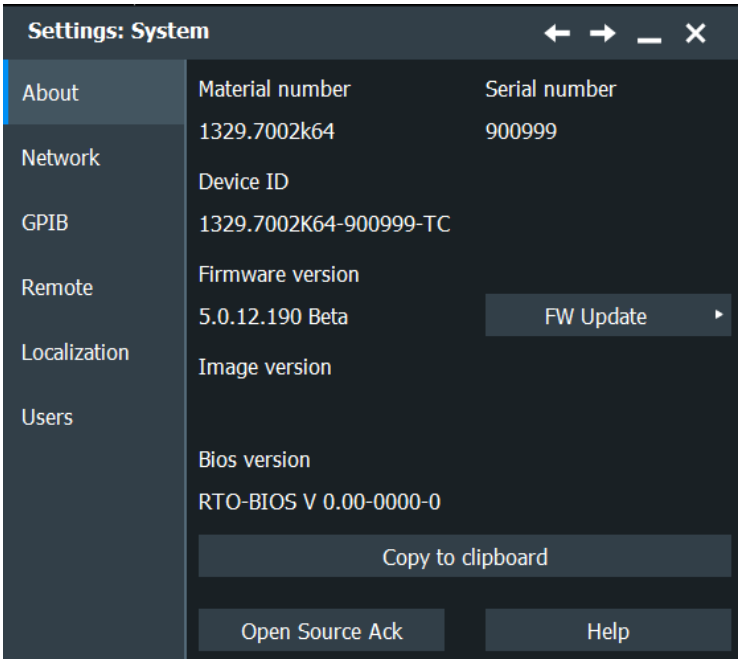
- System settings....67
- Option settings....71
- Appearance settings....74
- Display settings....90
- Frontpanel settings....94
- Preset setup....98
- Maintenance settings....100
- Save / recall....104

4.1 System settings

In the "Settings" > "System" dialog box, you find all instrument, firmware and network related information. Here you can also set the language that is used in the dialogs.

4.1.1 About settings

Access: "Menu" > "Settings" > "System" > "About"



FW Update
Opens the Firmware update dialog box.

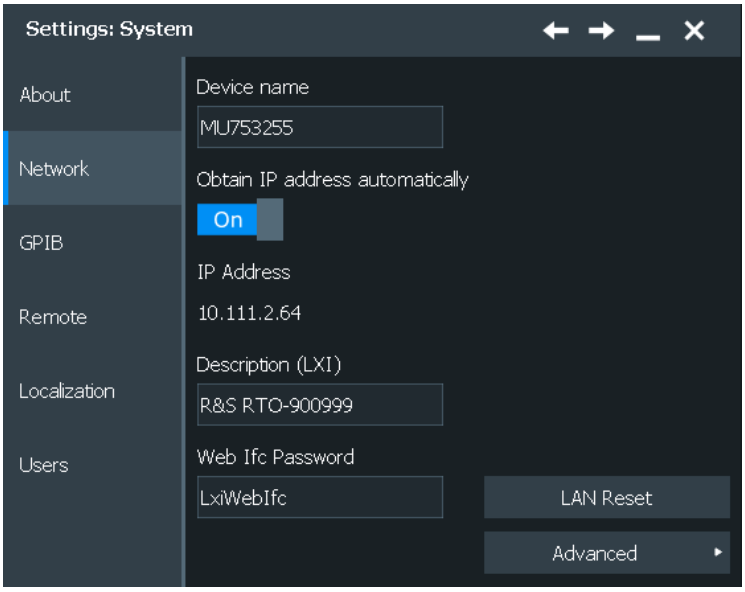
Copy to clipboard
Copies the instrument information, which you see in the dialog box, to the clipboard.

Open Source Ack
Opens the open source acknowledgment document, which provides verbatim license texts of the used open source software.

Help
Opens the help window with a table of contents, where you can browse and search for information.

4.1.2 Network settings

Access: "Menu" > "Settings" > "System" > "Network"



Device name
Indicates the currently defined computer name, the defined IP address and DHCP address enabling. These values are required to configure the instrument for work in a network.

NOTICE! Risk of network problems. All parameters can be edited here; however, beware that changing the device name has major effects in a network. For details, see Chapter 22.2, "Setting up a network (LAN) connection", on page 1228.

Remote command:
`DIAGnostic:SERVice:COMPutername` on page 1297

Obtain IP address automatically
If enabled, the IP address of the oscilloscope is obatained automatically.

IP Address
Indicates the currently defined computer name, the defined IP address and DHCP address enabling. These values are required to configure the instrument for work in a network.

Description (LXI)
Instrument description of the R&S RTO6.

Web Ifc Password
Password for LAN configuration. The default password is *LxiWebIfc*.

LAN Reset
Resets the LAN configuration to its default settings using the network configuration reset mechanism (LCI) for the instrument. The following parameters are reset:

Parameter	Value
TCP/IP mode	DHCP + auto IP address
Dynamic DNS	Enabled
ICMP ping	Enabled
Password for LAN configuration	LxiWebIfc

The LAN settings are configured using the instrument's web browser.

Advanced
Opens the standard Windows "Network Connections" dialog box to configure a network connection. Only users with administrator rights can fulfill this task.

4.1.3 GPIB settings

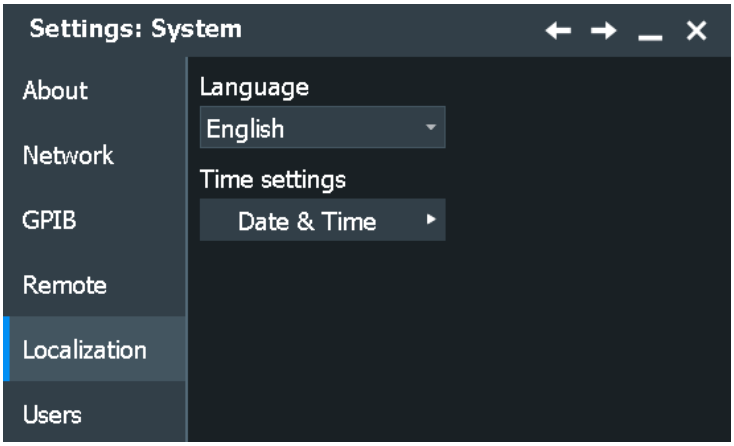
For a description of the GPIB settings, see: Chapter 22.5.2.2, "Remote settings", on page 1243.

4.1.4 Remote settings

For a description of the remote settings, see: Chapter 22.5.2.2, "Remote settings", on page 1243.

4.1.5 Localization settings

Access: "Menu" > "Settings" > "System" > "Localization"



Language
Selects the language in which the dialog boxes, result boxes and other screen information is displayed.

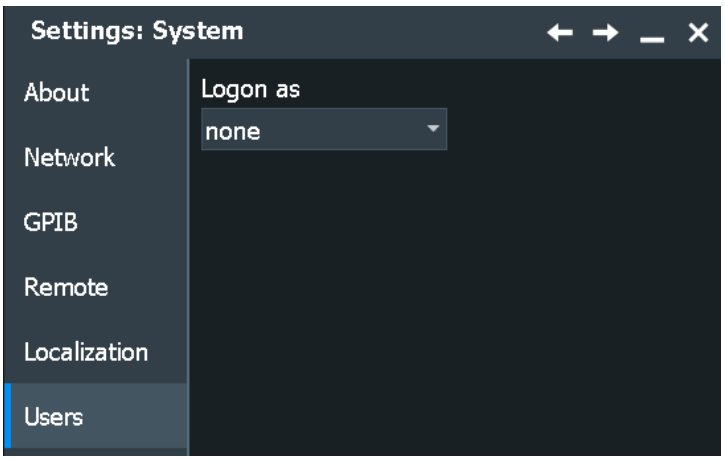
Time settings
Opens the standard Windows "Date and Time Properties" dialog box to set the correct date and time. Only users with administrator rights can fulfill this task.

Note: Usually date and time are set correctly. To adjust your regional time, select the correct time zone rather than changing the time.

Remote command:
`SYSTem:DATE` on page 1296
`SYSTem:TIME?` on page 1296

4.1.6 Users settings

Access: "Menu" > "Settings" > "System" > "Users"



Log on as
Sets the user that is automatically logged on during the startup process of the instrument. The change of this setting takes effect at the next instrument startup.

See Chapter 22.1.1, "Logon", on page 1224 for restrictions of the standard user and how to change the auto-logon.

"User autologon"	Auto-logon as standard user with limited access. Enter the "User name": *NormalUser* and the "Password" of the standard user.
"Admin autologon"	Auto-logon with unrestricted access to the instrument and network. The setting is only available for administrators. Enter the "User name": *Instrumen* and the administrator's "Password" to enable the auto-logon.
"None"	No auto-logon, user name and password are requested at instrument startup.

4.2 Option settings

Additional options for the R&S RTO6 are enabled using a license key. To obtain the license key, consult your sales representative.

The license type defines the duration of applicability and the portability of a license. The following license types are provided: evaluation, permanent, portable, quantified, timed with duration of 1, 3, 6 or 12 months. A license can also be in the states deactivated and expired.

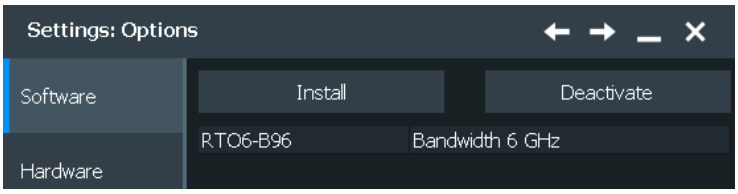
Unregistered licenses

Unregistered licenses are not assigned to a particular instrument. The instrument accepts only registered licenses. If your license is delivered unregistered, use the online tool R&S License Manager to register the license for your instrument. The registration of a permanent license is irreversible, so ensure that you register it for the correct instrument. The address of the tool is https://extranet.rohde-schwarz.com/service. For registration, you need the device ID of the instrument on which the option will be installed.

The R&S License Manager also allows you to move a portable license to another instrument.

4.2.1 Software options settings

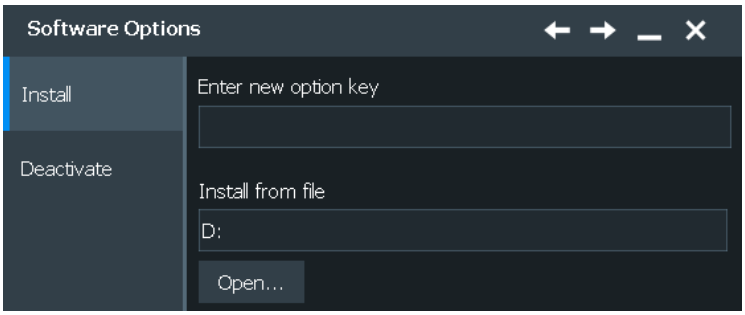
Access: "Menu" > "Settings" > "Options" > "Software"



In this dialog, you can access settings for installing and deactivating options.

4.2.1.1 Install options

Access: "Menu" > "Settings" > "Options" > "Software" > "Install"



In the "Install" tab, you can install new options or deactivate existing options using license keys.

Enter new option key

Enter the license key here to activate the option. For license keys delivered as a file, use Install from file. Only users with administrator rights can activate options.

Install from file
If you got a license file, install the license here.

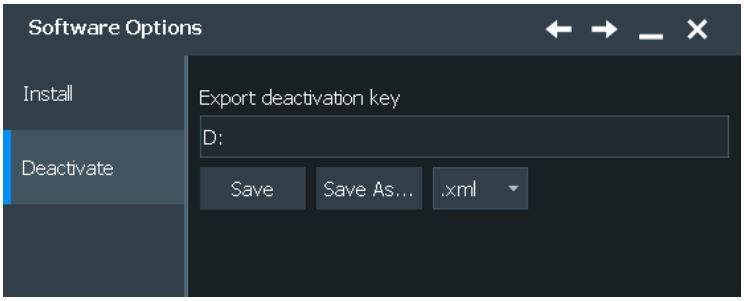
Tap "Open" to open the file selection dialog, or enter the complete path and filename. For details, see Chapter 12.5, "File selection dialog", on page 482. Only users with administrator rights can activate options.

When you move a portable license, use this function to import the deactivation key that is generated by the "R&S License Manager". See also Chapter 4.2.1.3, "Using a license server", on page 73.

4.2.1.2 Deactivate options

Access: "Menu" > "Settings" > "Options" > "Software" > "Deactivate"

In the "Deactivate" tab, you can export the deactivation response.



Export deactivation key
When you move a portable license, or deactivate an option, you have to note the response key, or to save the response to a file. The "R&S License Manager" needs the response key.

4.2.1.3 Using a license server

Software licenses can also be provided by the R&S®License Server. In this case, all available licenses are stored on a specified server. When you need a particular application, you can obtain a license from the server. When you no longer need the license, you return it to the server and it becomes available to other users again. These licenses are also referred to as *floating licenses*, as opposed to permanent licenses.

The R&S®License Server must be set up as described in the R&S®License Server - Managing Floating Licenses - User Manual.

You can find the manual in the Windows menu of your instrument: "Start > All Programs > R&S License Server > Open License Server Manual".

To open the License Server

► In the Windows notification area, select , and then "Open Manager".

a) Optional: If the status icon (, or) in the Windows notification area is not shown yet, click "Start > All Programs > R&S License Server > License Server Manager"".
The icon is now available in the Windows notification area.
b) Optional: If the status is (stopped), select the icon, and then select "Start License Server". Then, select > again.

The license server opens in the default web browser. The program adds all available Rohde & Schwarz products (smart cards or devices) to a list and automatically shows the first product that is detected. By default, the "Licenses" list opens with an initial filtering for showing only active licenses.

4.2.2 HW options settings

This tab informs about the availability of hardware options.

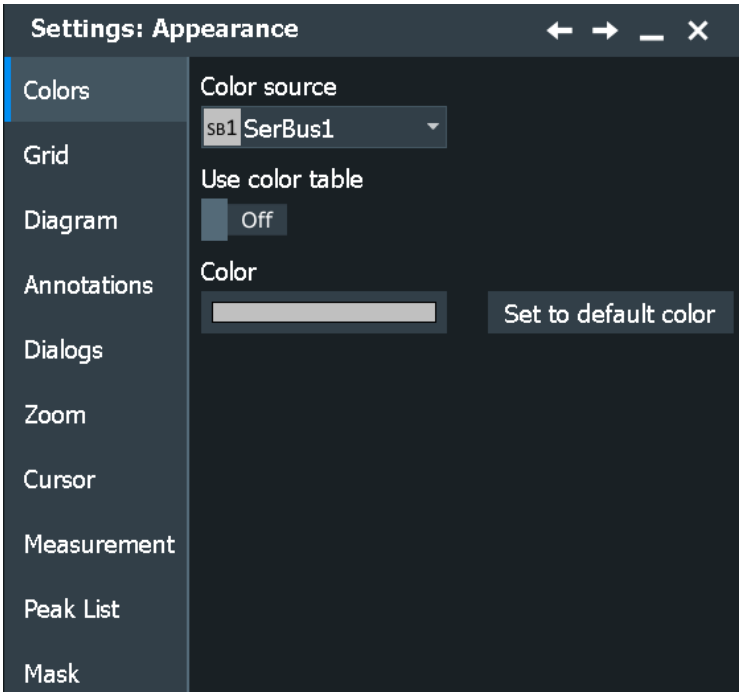
4.3 Appearance settings

In the "Settings" > "Appearance" dialog box, you define the look and feel of the display element, e.g. waveform colors, result position, or grid behavior.

4.3.1 Colors settings

Access: "Settings" > "Appearance" > "Colors"

In this dialog, you can define the colors to be used for the different waveforms.



Color source
Selects the waveform to which the color table and the labels are assigned.

A spectrogram (option R&S RTO6-K37) always has the same color as the math (spectrum) waveform from which it is created.

Use color table
If enabled, the selected waveform is displayed according to its assigned color table.

If this option is disabled, the selected color is displayed, and the intensity of the specific signal color varies according to the cumulative occurrence of the values.

Remote command:
`DISPlay:COLor:SIGNal<m>:USE` on page 1298

Color
Shows the current color of the selected waveform. To change the color, tap the button and select a color. The color of the waveform, its signal icon, channel icon, and of the illuminated keys is adjusted to the new color.

Remote command:
`DISPlay:COLor:SIGNal<m>:COLor` on page 1297

Assigned color table
Adjust the waveform colors to suit your preferences. For each of the following waveform types you can assign a suitable color table:
- Analog and digital channels
- Reference waveforms
- Results of a mathematical function, also for FFT and derived spectrogram.
- Measurements and tracks
- XY-traces

- Serial buses if a protocol option is activated
- Parallel buses if MSO option is installed

See also: Chapter 4.3.2, "Color tables", on page 76

Remote command:
`DISPlay:COLor:SIGNal<m>:ASSign` on page 1297

Set to default color
Resets the color of the selected waveform to the factory default.

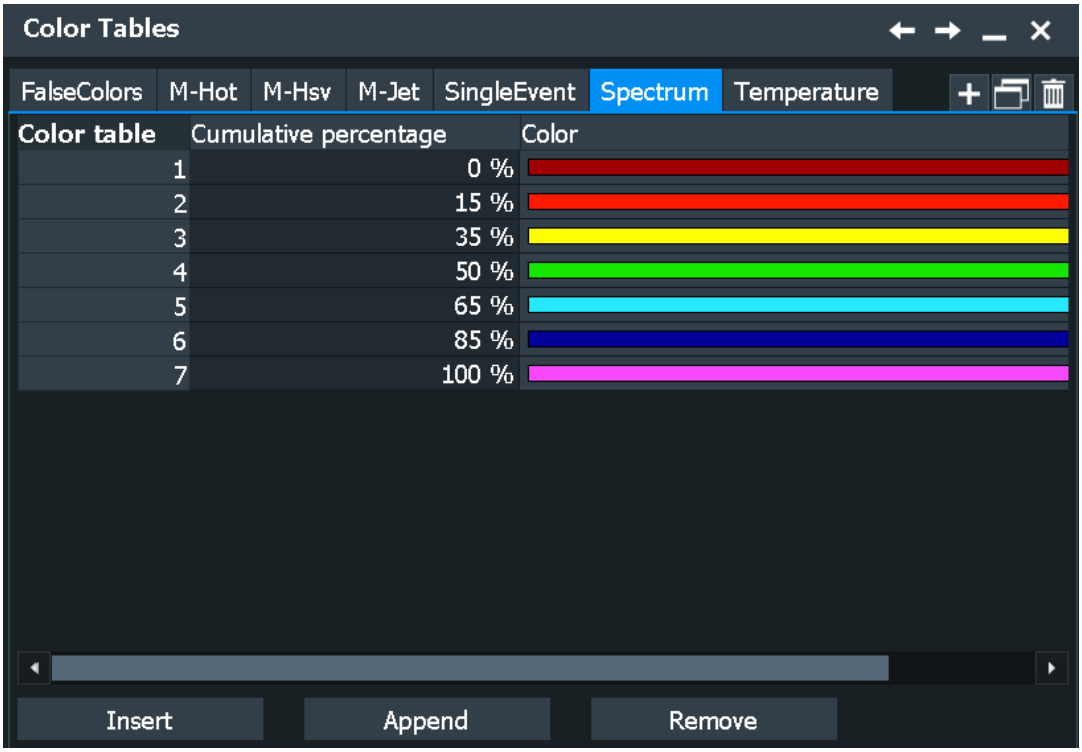
4.3.2 Color tables

Access: "Settings" > "Appearance" > "Color" > "Edit Color Table"

Color tables define the color of the waveform pixels depending on the cumulative occurrence of the associated values. By default, the intensity of the specific waveform color varies according to the cumulative occurrence of the values. The more often a value occurs, the darker the color of the data point is displayed.

The following color tables are provided:

- "False colors": color changes gradually in a wide color spectrum.
- "M-Hot"
- "M-Hsv"
- "M-Jet"
- "Spectrum": colors display the wave lengths of the light. Low cumulative occurrence is displayed blue like high wave length.
- "Single Event": single events and very seldom events appear yellow, a higher cumulative occurrence is shown with blue color. This view helps to identify specific events.
- "Temperature": color changes gradually from blue (low temperature) to red (high temperature) with increasing cumulative occurrence.



The editing table allows you to edit existing color tables or add new ones that can then be assigned to the waveforms.

See also:

- "Assigned color table" on page 75

Remote commands

The following remote commands are used to configure color tables:

`DISPlay:COLor:PALette:COUNt?` on page 1299

`DISPlay:COLor:PALette:ADD` on page 1298

`DISPlay:COLor:PALette:REMove` on page 1298

`DISPlay:COLor:PALette:POINt:INSert` on page 1299

`DISPlay:COLor:PALette:POINt:ADD` on page 1299

`DISPlay:COLor:PALette:POINt[:VALue]` on page 1299

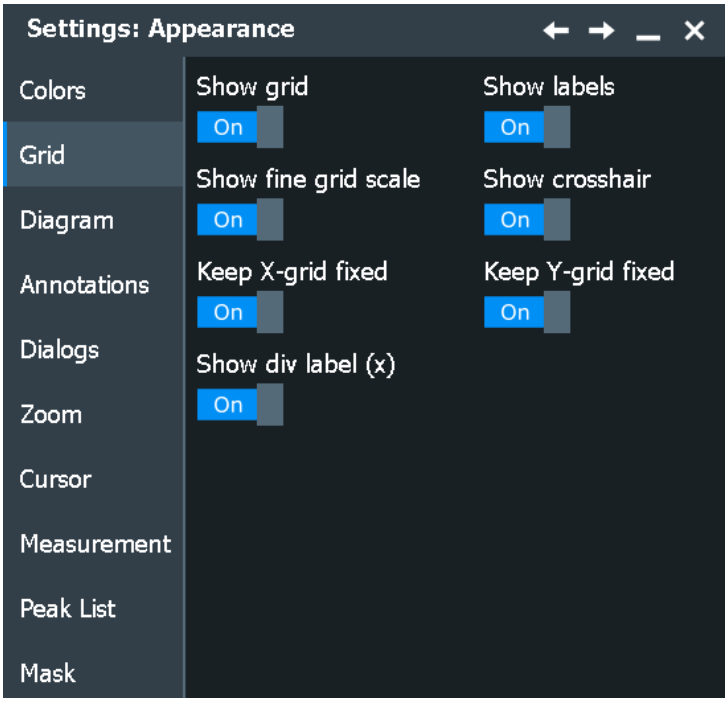
`DISPlay:COLor:PALette:POINt:COUNt?` on page 1300

`DISPlay:COLor:PALette:POINt:REMove` on page 1299

`DISPlay:COLor:PALette:COUNt?` on page 1299

4.3.3 Grid appearance settings

Access: "Settings" > "Appearance" > "Grid"



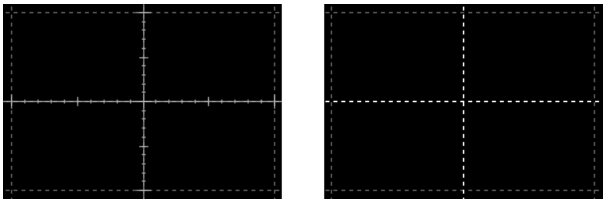
Show grid
If selected, a grid is displayed in the diagram area. A grid helps you associate a specific data point to its exact value on the x- or y-axis.

Remote command:
`DISPlay:DIAGram:GRID` on page 1300

Show labels
If selected, labels mark values on the x- and y-axes in specified intervals in the diagram.

Remote command:
`DISPlay:DIAGram:LABels` on page 1300

Show fine grid scale
If selected, the crosshair is displayed as a ruler with scale markers. If disabled, the crosshair is shown as dashed lines.



Remote command:
`DISPlay:DIAGram:FINegrid` on page 1301

Show crosshair
If selected, a crosshair is displayed in the diagram area. A crosshair allows you to select a specific data point by its co-ordinates.

Remote command:
`DISPlay:DIAGram:CROSshair` on page 1301

Keep X-grid fixed
If enabled, the vertical grid lines remain in their position when the horizontal position is changed. Only the values at the grid lines are adapted.

Remote command:
`DISPlay:DIAGram:XFIXed` on page 1301

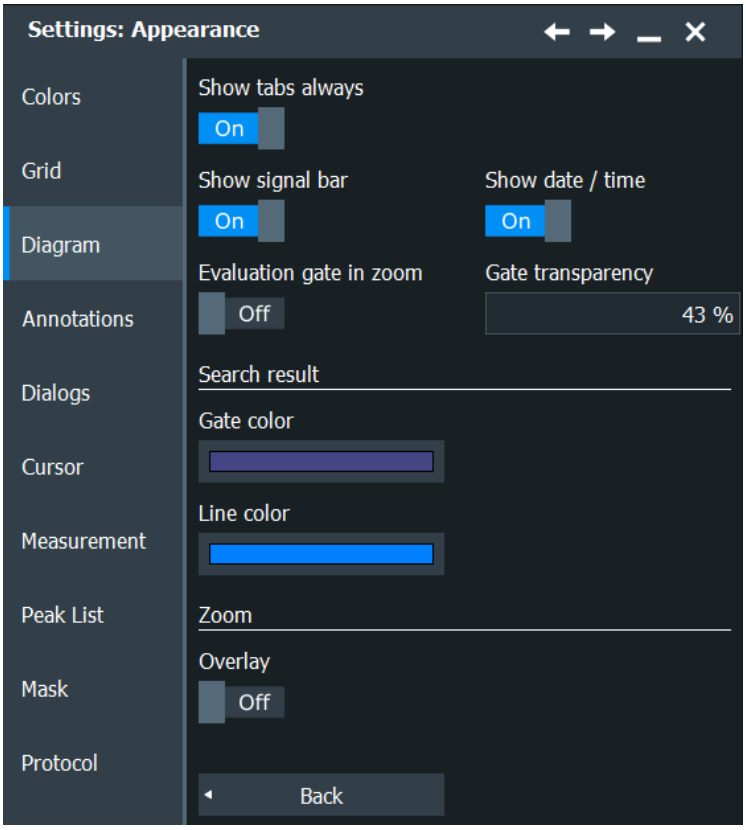
Keep Y-grid fixed
If enabled, the horizontal grid lines remain in their position when the position of the curve is changed. Only the values at the grid lines are adapted. Fixed horizontal grid lines correspond to the behavior of traditional oscilloscopes.

Remote command:
`DISPlay:DIAGram:YFIXed` on page 1301

Show div label (x)
If selected, the time scale value is shown at the diagram bottom instead of the horizontal grid labels. For example, 10 ns/div is shown instead of the values 0, 10, 20, 30... ns.

4.3.4 Diagram appearance settings

Access: "Settings" > "Appearance" > "Diagram" tab



Show tabs always
If selected, the tab titles of all diagrams are displayed: "Diagram1", "Diagram2" ...

If cleared, the tab titles are not shown except for titles in a tabbed diagram. In tabbed diagrams, the tab titles are required to change the tabs.

Remote command:
`DISPlay:DIAGram:TITLe` on page 1306

Show signal bar
If enabled, the signal bar is displayed on the right of the diagram area.

Remote command:
`DISPlay:SIGBar[:STATe]` on page 1306

Show date / time
If enabled, the date and time are shown near to the Rohde & Schwarz logo at the upper right corner of the screen.

Evaluation gate in zoom
If enabled, the available histogram areas, masks, and measurement gates are shown in the zoom diagrams. If the evaluation gate is within the zoom area, the display helps to move or modify the evaluation gates in the zoom window.

Make sure that the option is disabled if the zoom area and the evaluation gate are of nearly the same size to avoid conflicts in operation.

Gate transparency
Sets the transparency of the area that is defined as measurement or search gate. The setting only takes effect if "Show gate" is enabled.

Remote command:
`DISPlay:GATE:TRANsparency` on page 1306

Search result gate symbol color
Sets the color of the search zoom area. The search zoom area is displayed if "Show search zoom windows" is enabled.

See also: Chapter 11.4.2, "Display zoom settings", on page 440.

Search result line color
Sets the color of the search result markers. The markers are displayed if "Show search zoom windows" is enabled.

Zoom overlay
Shows all zooms of a diagram in one zoom window. The zoomed areas are overlaid for better comparison of the zoomed waveforms.

The setting affects all zoom diagrams.

See also: Chapter 7.1, "Zoom", on page 239

Remote command:
`LAYout:ZOOM:ONEDiagram` on page 1307

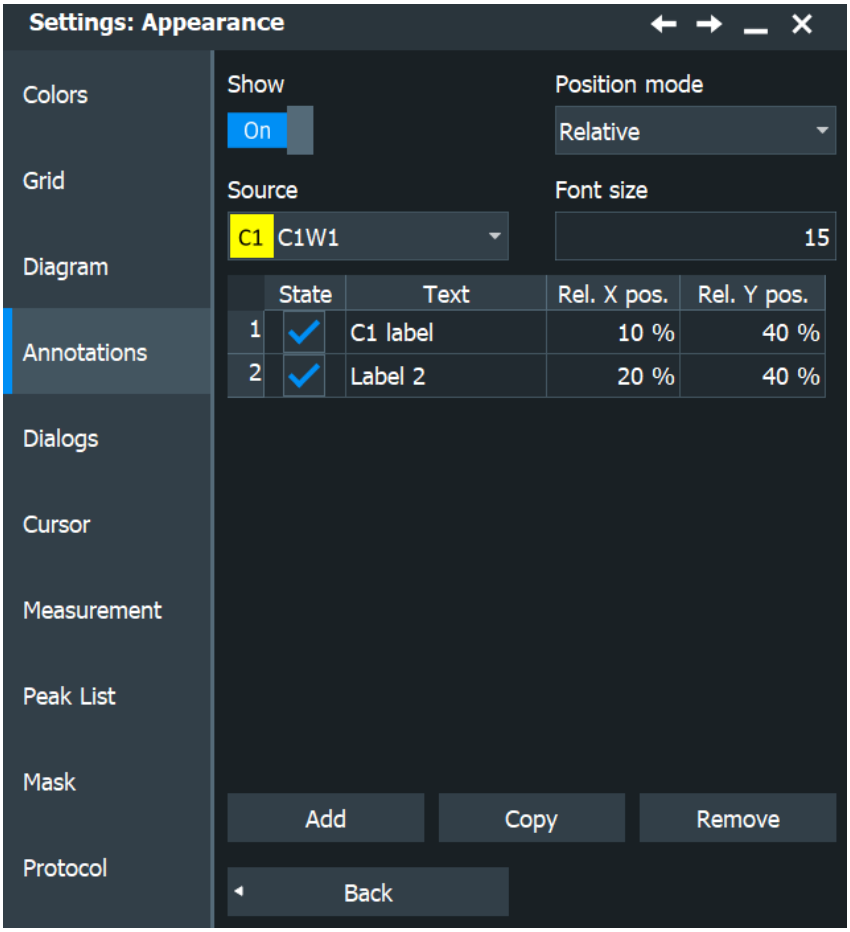
4.3.5 Annotations appearance settings

Access: "Settings" > "Appearance" > "Annotations" tab

Using labels, you can annotate the waveforms to name or explain each waveform. The text is shown in the same color as the assigned waveform. Each label has its individual position.

You can enter exact positions in the dialog box, or drag the labels on the screen to the required position. The position can be a fixed one (relative to the screen), or a flexible position (absolute, assigned to the axes).

To add labels quickly, you can add the "Label" icon to the toolbar and use it.



Show / Source

Selects the souce of the label and enables the display.

Remote command:

`DISPlay:SIGNal:LABel:SHOW` on page 1310

Position mode

Defines the label position either relative to the diagram or with absolute values according to the units of the waveform. Relative positions are fixed, whereas absolute positions move with the waveform display when the scales, the vertical position or offset, or the reference point are changed.

The position mode applies to all labels of the selected waveform. For different waveforms, different position modes can be selected.

"Relative" Sets a fixed position in percent of the screen counting from the upper left corner.

"Absolute" Sets the position in time and voltage values, or in other units depending on the waveform character.

Remote command:

`DISPlay:SIGNal:LABel:POSMode` on page 1307

Font size
Defines the size of the labels in the diagram.

Remote command:
`DISPlay:SIGNal:LABel:FONTsize` on page 1308

Labels
For each waveform, the "Labels" table shows the defined labels with their visibility state, label texts and positions. The labels belog to the selected waveform.

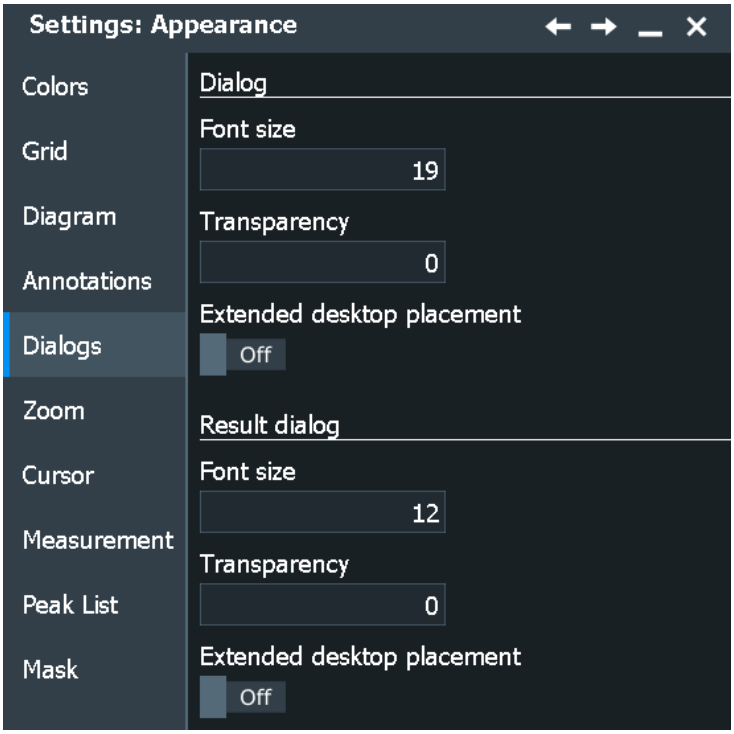
Enter the label text and the horizontal and vertical positions for each label. Enable "State" to show the label on the screen. The individual state of the labels takes effect if "Show" is active.

"Add"	Adds a line at the end of the list.
"Copy"	Copies the selected line in a new line.
"Remove"	Deletes the selected line. Only single lines can be removed. You can also delete a label by using the toolbar: Tap the "Delete" icon and then the label.

Remote command:
`DISPlay:SIGNal:LABel:ADD` on page 1308
`DISPlay:SIGNal:LABel:STATe` on page 1311
`DISPlay:SIGNal:LABel:IDSList` on page 1310
`DISPlay:SIGNal:LABel:REMove` on page 1309
`DISPlay:SIGNal:LABel:TEXT` on page 1312
`DISPlay:SIGNal:LABel:HORizontal:ABSolute:POSition` on page 1312
`DISPlay:SIGNal:LABel:VERTical:ABSolute:POSition` on page 1312
`DISPlay:SIGNal:LABel:HORizontal:RELative:POSition` on page 1313
`DISPlay:SIGNal:LABel:VERTical:RELative:POSition` on page 1313

4.3.6 Dialogs appearance settings

Access: "Settings" > "Appearance" > "Dialogs" tab



Dialog
With these settings, you can change the appearance of the dialogs.

Font size ← Dialog
Defines the font size of the text in dialog boxes.

Dialog transparency ← Dialog
Defines the transparency of the dialog box background. For high transparency values, you can see the waveform display in the background, and possibly check the effect of the changed setting. For lower transparency values, readability in the dialog box improves.

Result dialog
With these settings, you can change the appearance of the result dialogs.

Font size ← Result dialog
Defines the font size of the text in result boxes. The size of the result box is adapted to the font size.

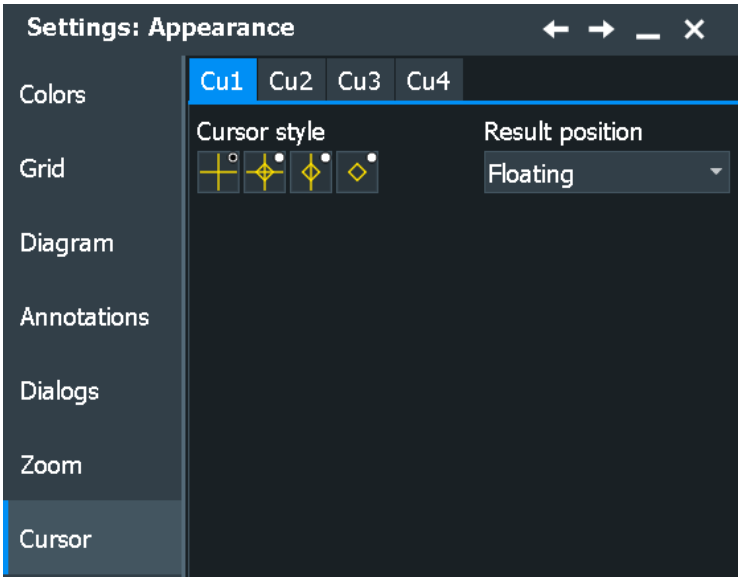
Transparency ← Result dialog
Defines the transparency of the measurement result boxes in the same way as Dialog transparency.

Extended desktop placement
If an external monitor is connected to the instrument, you can enable these settings to display dialog boxes and/or result boxes on the external monitor. Thus, the boxes do not cover the waveforms on the instrument display.

Remote command:
`DISPlay:EXTended:PORDialogs` on page 1313
`DISPlay:EXTended:POSDialogs` on page 1313

4.3.7 Cursor appearance settings

Access: "Settings" > "Appearance" > "Cursor" tab



Cursor style
Defines how the cursor is displayed in the diagram.

"Lines"	The cursors are displayed as lines.
"Line & Rhombus"	The cursors are displayed as lines. The intersections of the cursors with the waveforms are displayed by rhombus-shaped points.
"Vertical line and rhombus"	The cursors are displayed as vertical lines. The intersections of the cursors with the waveforms are displayed by rhombus-shaped points.
"Rhombus"	The intersections of the cursors with the waveforms are displayed by rhombus-shaped points.

Remote command:
`CURSor<m>:STYLe` on page 1314

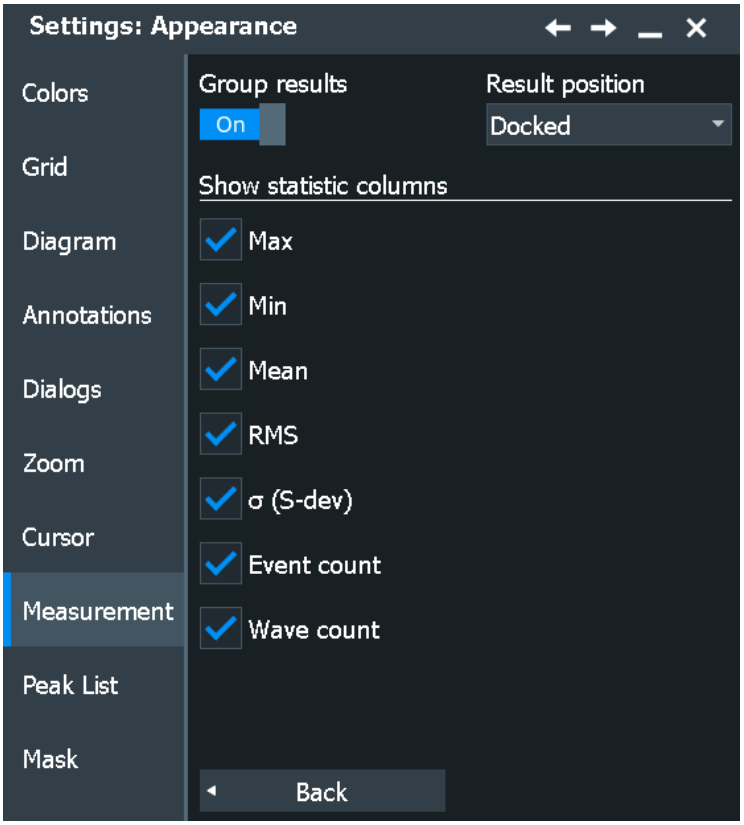
Result position
Defines the position of the cursor measurement results.

"Floating"	Floating result box in front of the diagrams.
"Preview"	Result icon on the signal bar.
"Docked"	Fixed tab below the diagrams.

Remote command:
`DISPlay:RESultboxes:CUPosition` on page 1314

4.3.8 Measurement appearance settings

Access: "Settings" > "Appearance" > "Measurement" tab



Group results
If enabled, all results are shown in one result box (default).

If disabled, the results of each measurement group are shown in a separate result box. The default position is ignored.

The setting affects all measurements except for the peak list.

Remote command:
`MEASurement<m>:DISPlay:GROuping` on page 1315

Result position
Defines the default position of the result table, when a new measurement is started:
- "Docked": fixed tab below the diagrams
- "Preview": result icon on the signal bar
- "Floating": floating result box in front of the diagrams

The setting affects only grouped results.

Remote command:
`DISPlay:RESultboxes:MEPosition` on page 1315

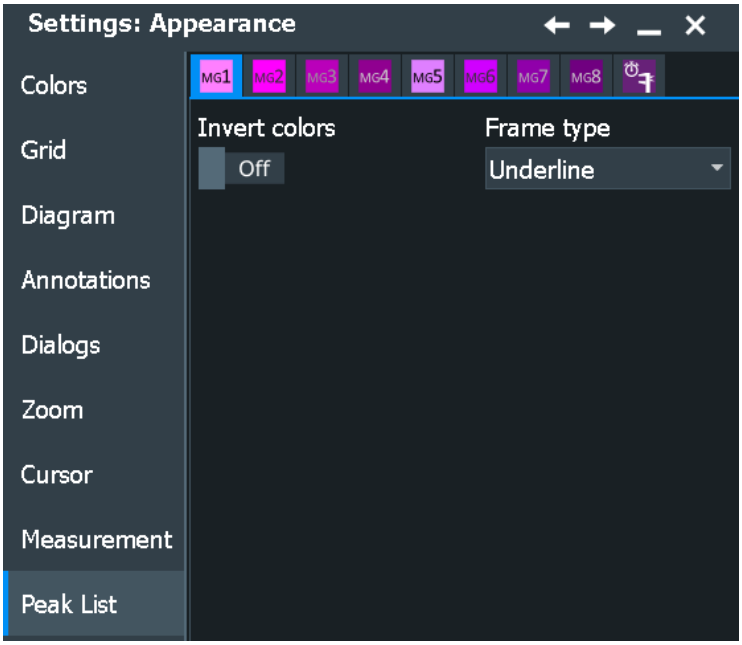
Show statistic columns
Select the statistical values that you want to see in the results table.

4.3.9 Peak list appearance settings

Access: "Settings" > "Appearance" > "Peak List" tab

For peak lists measurements only, requires option R&S RTO6-K37. You can enable the "Peak list" for "Spectrum" measurements, see Chapter 8.2.7, "Spectrum measurements", on page 334.

In this dialog, you can configure the look of the labels for peaks in the spectrum diagram.



Frame type
Defines the layout of the labels (full border, underline, or none).

Remote command:
`MEASurement<m>:RESult:LABorder` on page 1315

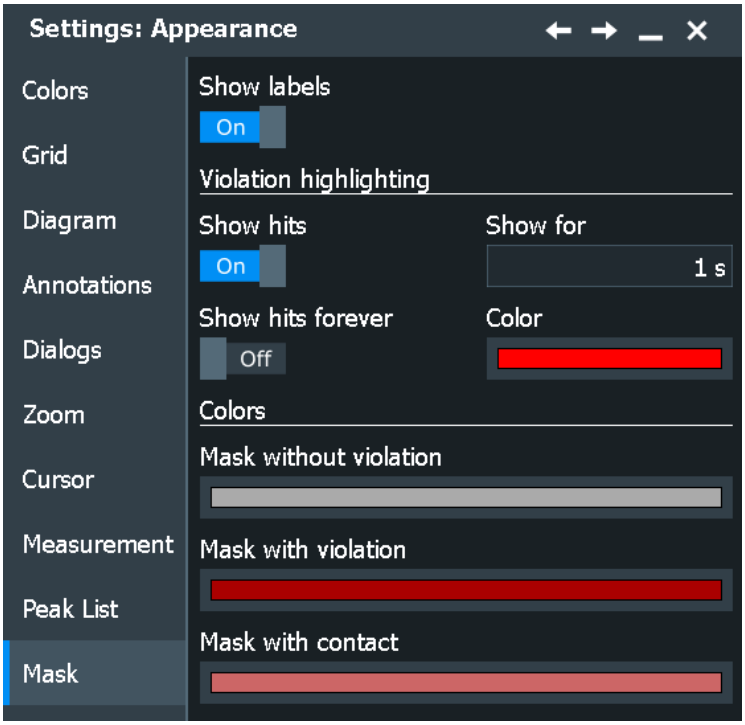
Invert
Displays black font on white background using the "Full border" frame type.

Remote command:
`MEASurement<m>:RESult:INVerse` on page 1316

4.3.10 Mask appearance settings

Access: "Settings" > "Appearance" > "Mask"

In this dialog, you can define the settings for mask and hit display. See also Chapter 10, "Mask testing", on page 394.



Show labels
Switches the display of the mask test name on or off.

To change the name of the mask test, open the "Mask Test" dialog > "Define Mask" tab. Double-tap the mask test subtab and enter the new mask test name.

Remote command:
`MTESt:LABel` on page 1317
`MTESt:REName` on page 1317

Show hits
If selected, the mask hits are highlighted on the screen. You can define the color and the time of the hit display.

These settings are not applied to trigger zones, which are based on masks.

Remote command:
`MTESt:HIGHlight:STATe` on page 1317

Show for
Sets the time for which the mask hits are shown.

Remote command:
`MTESt:HIGHlight:TIME` on page 1317

Show hits forever
If selected, the mask hits are shown for an unlimited period of time.

Remote command:
`MTESt:HIGHlight:INFinite` on page 1317

Color
Sets the color of samples that violated the mask.

Remote command:
`MTESt:COLor:MATCh` on page 1318

Mask without violation
Sets the color of masks segments that were not hit.

Remote command:
`MTESt:COLor:UNMatch` on page 1318

Mask with violation
Sets the color of mask segments the signal has entered into.

Remote command:
`MTESt:COLor:INTerior` on page 1319

Mask with contact
Sets the color of masks segments that were touched at the border. In this case, the resolution is not sufficient to detect if the mask was hit or not. Zoom into the concerned area to see the actual result.

Remote command:
`MTESt:COLor:BORDer` on page 1318

4.3.11 Protocol appearance settings

Access: "Settings" > "Appearance" > "Protocol"

In this dialog, you can define the position of the result table for the protocol analysis.

See also Chapter 13, "Protocol analysis", on page 484.

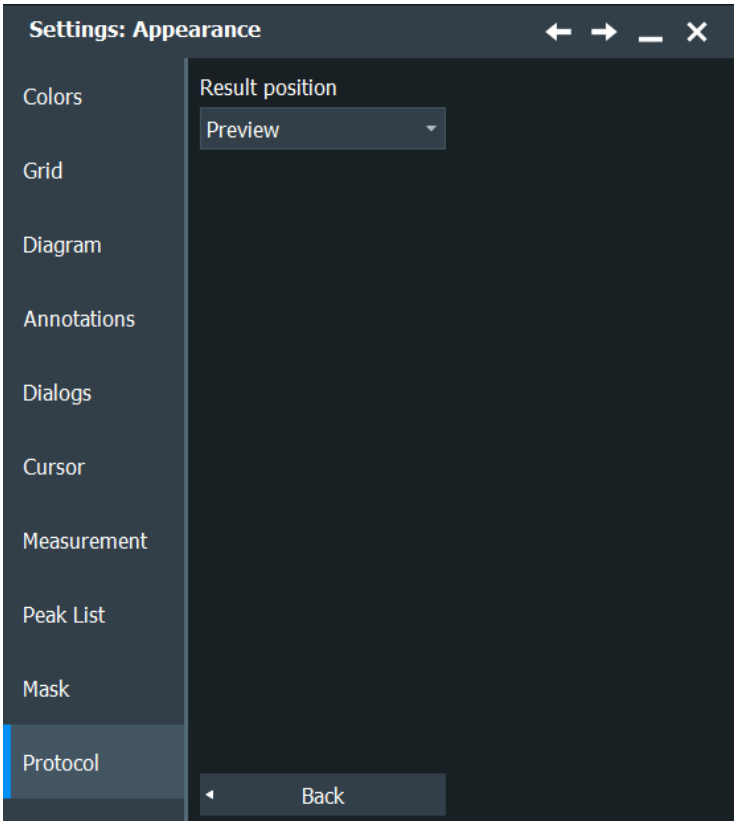


Table position
Defines the position of the decode table on the screen.

"Floating"	Floating result box in front of the diagrams.
"Preview""	Result icon on the sidebar.
"Docked"	Fixed tab below the diagrams.

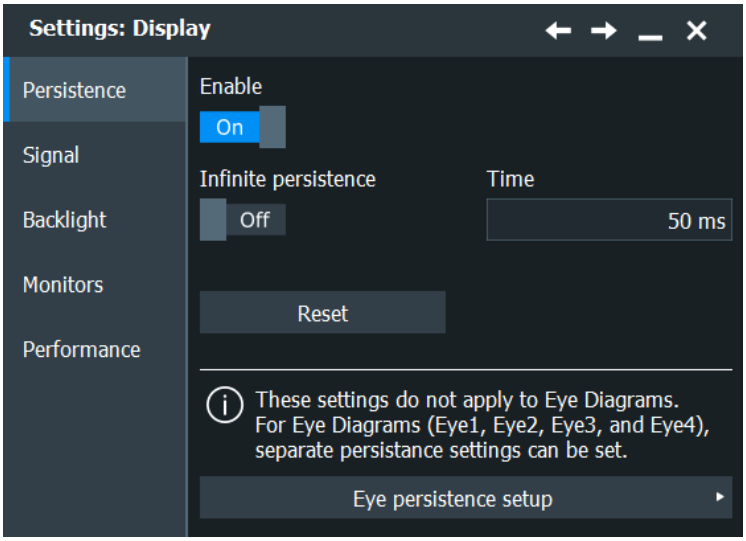
Remote command:
`DISPlay:RESultboxes:DEPosition` on page 1673

4.4 Display settings

In the "Settings" > "Display" dialog box, you can define the display settings like brightness and signal intensity. Yo can also set up an external monitor.

4.4.1 Persistence settings

Access: "Menu" > "Settings" > "Display" > "Persistence"



Enable
If enabled, each new data point in the diagram area remains on the screen for the duration that is defined using Time, or as long as Infinite persistence is selected.

If disabled, the waveform points are displayed only for the current acquisition.

Remote command:
`DISPlay:PERSistence[:STATe]` on page 1319

Infinite persistence
If infinite persistence is enabled, each new waveform point remains on the screen until this option is disabled. Use infinite persistence to display rare events in the signal.

Remote command:
`DISPlay:PERSistence:INFinite` on page 1320

Time
Sets a time factor that controls how long the waveforms points fade away from the display. Thus, the R&S RTO6 emulates the persistence of analog phosphor screens.

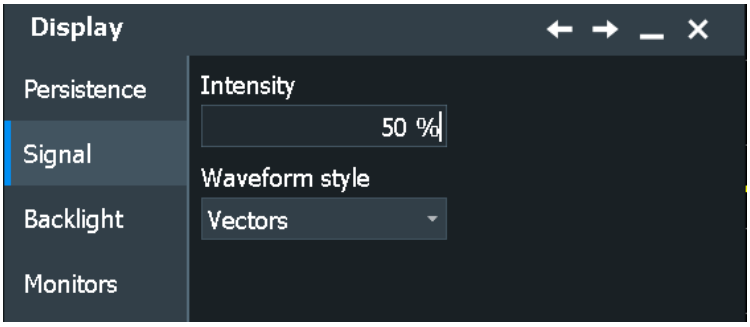
Remote command:
`DISPlay:PERSistence:TIME` on page 1320

Reset
Resets the display, removing persistent all waveform points.

Remote command:
`DISPlay:PERSistence:RESet` on page 1320

4.4.2 Signal settings

Access: "Menu" > "Settings" > "Display" > "Signal"



Intensity
This value determines the strength of the waveform line in the diagram. Enter a percentage between 0 (not visible) and 100% (strong). The default value is 50%.

You can also use the [Intensity] knob on the left side of the screen to adjust the waveform intensity directly.

Note: Use of color tables. The exact mapping of the cumulative value occurrences according to the assigned color table is guaranteed only if the intensity is set to 50%. All other intensity values falsify the mapping but can improve the visibility of the signal.

Remote command:
`DISPlay:INTensity` on page 1320

Waveform style
Select the style in which the waveform is displayed:

"Vectors"	The individual waveform points are connected by a line. Define the strength of the line using the [Intensity] knob.
"Dots"	Only the individual waveform points are displayed. Waveform sample points are the ADC sample points and additional interpolated points if "Interpolated time" is used for resolution enhancement. To see the dots of one waveform, perform one acquisition with [Single] and N=1 ("Average count" = 1). During continuous acquisition, or a [Single] acquisition with N > 1, the dots of multiple subsequent waveforms are displayed on the screen, and the waveform looks like a line. Consider also the "Interpolation" on page 128.

Remote command:
`DISPlay:DIAGram:STYLe` on page 1321

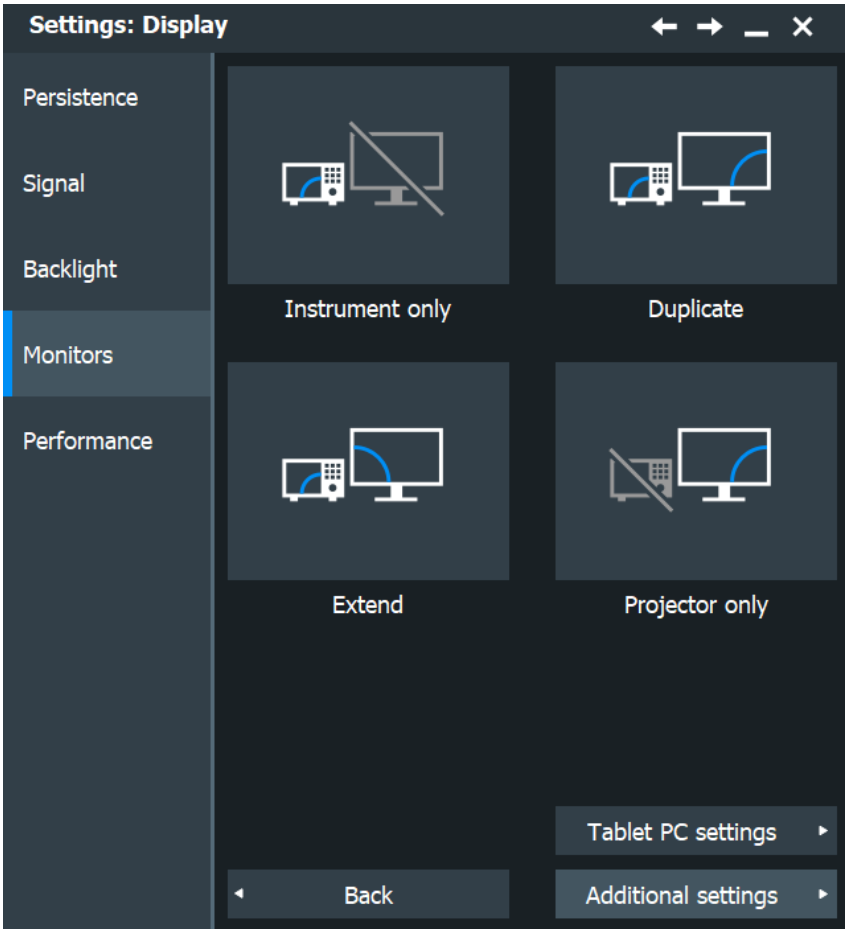
4.4.3 Backlight settings

Access: "Menu" > "Settings" > "Display" > "Backlight"

Brightness
Changes the background luminosity of the touchscreen.

4.4.4 Monitors settings

Access: "Menu" > "Settings" > "Display" > "Monitors"



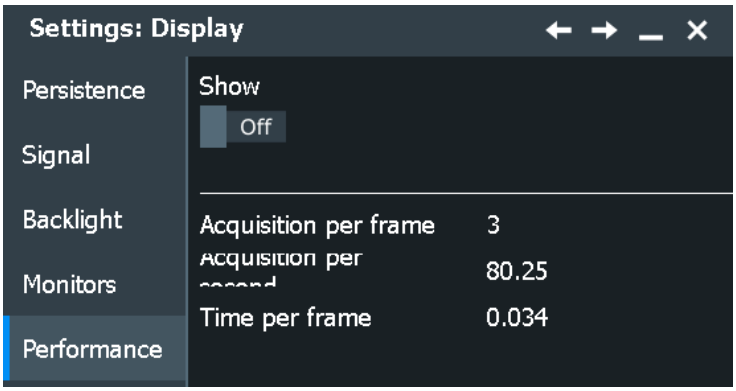
In the "Monitors" tab you can extend or duplicate the instrument display to a second monitor or projector (external display). The following options are available:

- "Instrument only": the instrument display is on, the external display is off.

- "Duplicate": the external display shows the same content as the instrument display.
- "Extend": the instrument display and the external display show different content.
- "Projector only": the instrument's user interface is only shown on the external display, the instrument display is off.
- "Tablet PC settings": opens the Windows setup for touch display to adjust the touch functionality when an additional monitor is in use.
- "Additional settings": opens the Windows configuration for display settings.

4.4.5 Performance settings

Access: "Menu" > "Settings" > "Display" > "Performance"



Show
Displays the "Performance" results box.

The "Performance" result box shows information on the current acquisition performance values of the R&S RTO6.

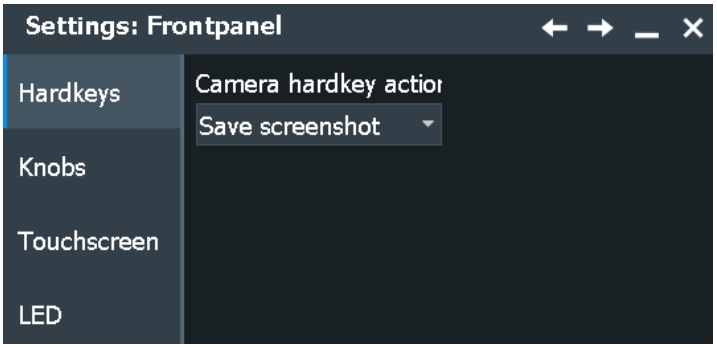
The instrument groups acquired waveforms together in a frame, and displays the frame content. The maximum number of frames displayed per second is about 30. The current number of frames per second is indicated as reciprocal "Time per frame". If the time scale decreases, and thus the number of acquisitions per second also decreases, the number of acquisitions per frame can drop to 1.

4.5 Frontpanel settings

In the "Frontpanel" dialog box, you can assign functions to keys and knobs, adjust the Multiuse knob and adjust the brightness of the keys.

4.5.1 Hardkeys: function assignment

Access: "Settings" > "Frontpanel" > "Hardkeys"



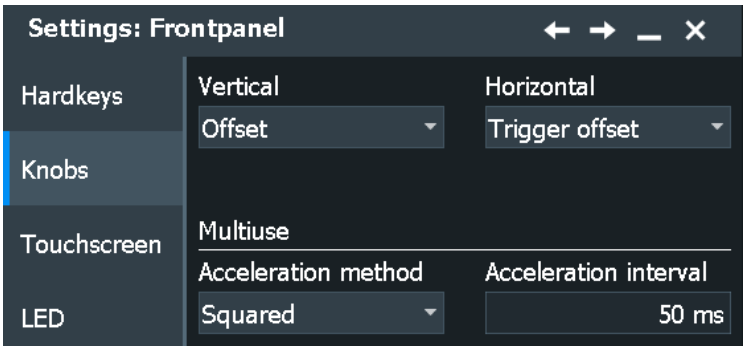
Camera hardkey action
The Camera key on the left side of the display is a shortcut key that initiates an associated action.

You can assign one of the following actions:
- Save a screenshot
- Open screenshot setup
- Save a report
- Open report setup

Configure the settings for the selected action.
- Screenshots: "Save/Recall" key > "Save" tab > "Screenhot", see Chapter 12.3.1, "Screenshot settings", on page 476.
- Report: "Save/Recall" key > "Save" tab > "Report", see Chapter 12.4.1, "Report settings", on page 480.

4.5.2 Knobs

Access: "Settings" > "Frontpanel" > "Knobs"



Vertical 95
Horizontal 96
Acceleration method 96
Acceleration interval 96

Vertical
Selects the action that you want to perform with the vertical [Position] knob. It can change the offset or the waveform position of the selected waveform.

See also: "[Position]" on page 38.

Horizontal
Selects the action that you want to perform with the horizontal [Position] knob. It can change the horizontal position of the trigger point (trigger offset), or the reference point.

See also: "[Position]" on page 36.

Acceleration method
Selects a method to accelerate the movement of the element on the screen compared to the actual movement of the [Multiuse] knob.

Acceleration is useful if you need to move from one end of the screen to the other, for example. Without acceleration, you have to turn the knob quite a while to reach the other end. But acceleration can make precise selection difficult, because a small movement of the knob causes a relatively large movement on the screen.

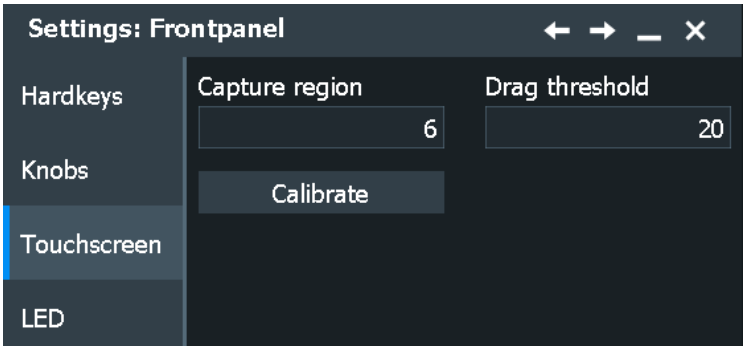
"None"	No acceleration method used.
"Squared"	Moderate acceleration method used.
"Exponential"	Strong acceleration method used.

Acceleration interval
Defines the delay time during which the movement of the [Multiuse] knob is analyzed before acceleration is applied. For short intervals, acceleration sets in quickly, but is not as effective. For long intervals, acceleration is more effective. However, it takes longer until the instrument reacts on the knob's input.

Furthermore, when you turn the knob slowly during fine-tuning, subsequent movements that occur during the same interval are accelerated, making precise selection difficult.

4.5.3 Touchscreen

Access: "Settings" > "Frontpanel" > "Touchscreen".



Capture region
Defines the number of pixels around each element (e.g. button, icon, data point) that create a capture region. If you tap your finger or click the mouse pointer within this capture region, this element is considered to be selected. If you tap or click outside this area, a different or no element is selected.

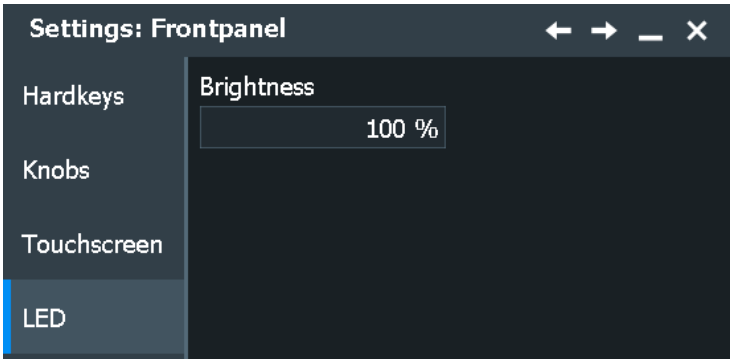
The larger the region, the easier is it to select an element. However, when selecting data points, for example, a large frame does not allow you to select precisely.

Drag threshold
Defines the maximum number of pixels around an element (e.g. data point) within which your pointing device must stay to "click" the element. When you tap or click a specific element and move your finger or the mouse outside this range, it is considered to be a "dragging" or "moving" operation.

Calibration
Opens the touchscreen calibration application. Only users with administrator rights can fulfill this task.

4.5.4 LED

Access: "Settings" > "Frontpanel" > "LED".



Brightness
Defines the luminosity of illuminated front panel keys and knobs.

4.5.5 Aligning the touchscreen

When the device is delivered, the touchscreen is initially calibrated. However, to ensure that the touchscreen responds to the finger contact correctly, a touchscreen alignment is required. Only users with administrator rights can fulfill this task.

Alignment of the touchscreen is useful:

- At first use
- If the position of the instrument has been changed, and you cannot look straight on the screen
- If another person operates the instrument
- If you notice, that touching a specific point on the screen does not achieve the correct response

1. In the "Menu" > "Settings" menu, select "Frontpanel".

2. Select the "Touchscreen" tab.
3. Tap "Calibrate".

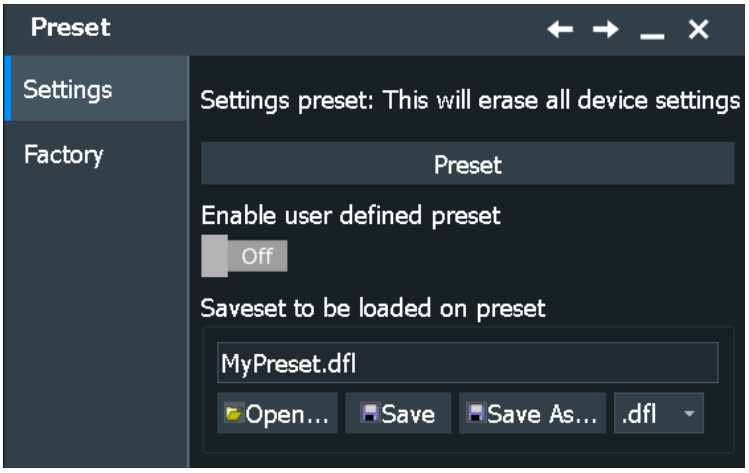
   A blinking cross appears in the lower left corner of the screen.
4. Touch and hold the cross until "OK" is shown.
5. Repeat this action for the crosses in the other corners.
6. Tap the Rohde & Schwarz logo to display the instrument's user interface.

4.6 Preset setup

A user-defined preset contains the complete instrument setup including display settings, except for transparency and intensity. You can save the current configuration to a preset file, and load a previously saved preset file. You can then specify that these settings are to be applied with the [Preset] key.

4.6.1 User-defined preset

Access: "Menu" > "Settings" > "Preset" > "Settings" tab



Preset
Performs a user-defined preset of the instrument.

Saveset to be loaded on preset
The file name with extension `.dfl` to load or to save the settings to.

Remote command:
`USRDefined:RST:NAME` on page 1321
`USRDefined:RST:SAVE` on page 1322
`USRDefined:RST:OPEN` on page 1321

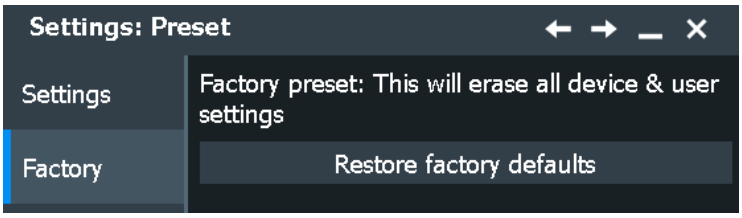
Enable user defined preset
If enabled, the settings from the selected preset file are restored when the [Preset] key is pressed.

If disabled, [Preset] sets the instrument to the factory defaults.

Remote command:
`USRDefined:RST[:ENABle]` on page 1322

4.6.2 Factory preset

Access: "Menu" > "Settings" > "Preset" > "Factory"



Restore factory defaults
Resets the instrument to the factory default settings, to the initial state. Factory settings comprise all instrument settings, including display, intensity and transparency settings. After loading factory defaults, perform a self-alignment to synchronize the signal data.

Remote command:
Chapter 24.7.1, "System", on page 1296

4.6.3 Restoring settings

When you have changed many different settings on the instrument and are no longer sure which settings are causing which effect in the measurement, you may want to restore the default settings and start anew. The following methods are available:

- Saving instrument settings to a user-defined preset and restoring the instrument settings to user-defined default values
- Restoring all settings on the R&S RTO6 to the factory-defined values
- Restoring settings from a file (see "To load settings from a saveset file" on page 451)

For details on save/recall instrument settings and associated remote commands, see Chapter 12.1.1, "Savesets", on page 445.

To save a user-defined preset

1. Open the "Menu" > "Settings" > "Preset" tab.
2. Enter a name for the preset file. Select the file format.
3. Tap Save.

Note: If you want to store the file in another directory than the default one, select "Save As". See also: Chapter 12.5, "File selection dialog", on page 482

To restore the instrument settings to user-defined default values

1. Open the "Menu" > "Settings" > "Preset" tab.
2. Tap "Open" and select the preset file that contains the required settings.

   The instrument settings are restored to values that are stored in the file.
3. To use these settings as preset values, select "Enable user-defined preset".
4. Press the [Preset] key.

To restore all settings to the factory defaults

1. Open the "Menu" > "Settings" > "Preset" > "Factory " tab.
2. Tap the "Restore factory defaults" button.

   All settings on the R&S RTO6 are reset to their factory-defined values. As long as no user-defined preset file is loaded and "Enable user defined preset " is disabled, the [Preset] key also resets the instrument settings to factory defaults.

4.7 Maintenance settings

In the "Settings" > "Maintenance" dialog box, you define the the look and feel of the display element, e.g. waveform colors, result position, or grid behavior.

4.7.1 Firmware update

Access: "Menu" > "Settings" >"Maintenance" > "FW Update"

Your instrument is delivered with the latest firmware version. Firmware updates are provided on the internet at:

www.rohde-schwarz.com/firmware/rto6.

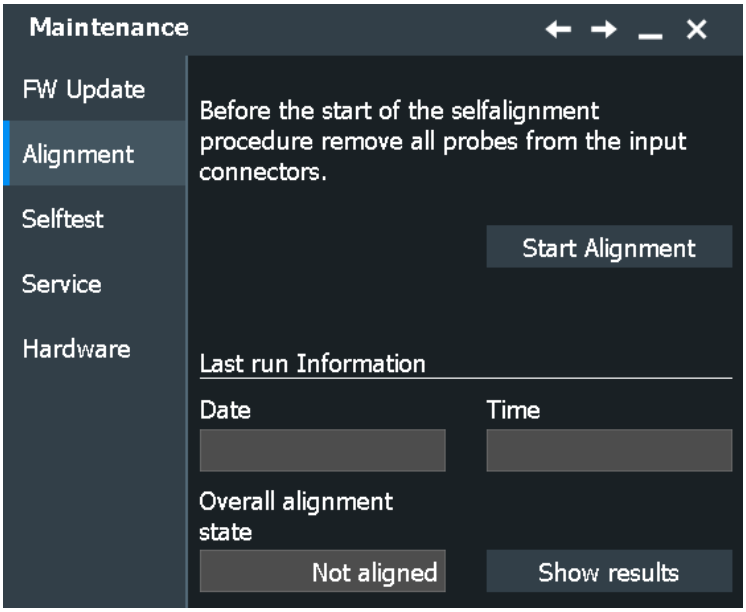
The "Release Notes" describe the improvements and modifications of all firmware versions. They also explain how to update the firmware. They are available along with the firmware on the same web page.

4.7.2 Alignment

When data from several input channels is displayed at the same time, it may be necessary to align the data vertically or horizontally to synchronize the time bases or amplitudes and positions. This is the case, for example, when strong temperature changes occur (> 5°).

4.7.2.1 Control

Access: "Menu" > "Settings" >"Maintenance" > "Alignment"



Start Alignment
Starts the self-alignment procedure for all channels.

Remote command:
`*CAL?` on page 1288

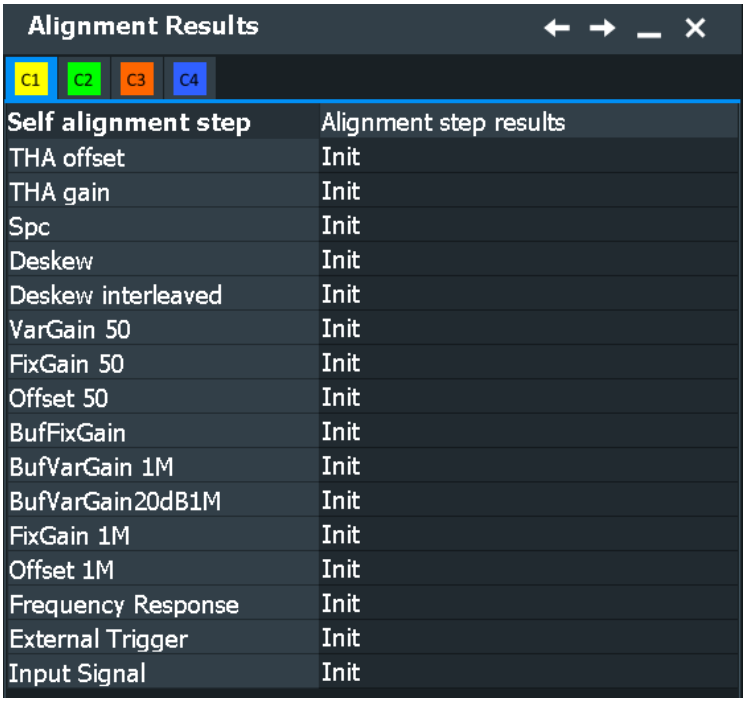
Date / Time / Overall alignment state
Show the date, time and the summary result of the self-alignment process: not aligned, passed or failed. Detailed results are provided on the "Results" tab.

Remote command:
`CALibration:DATE?` on page 1322
`CALibration:TIME?` on page 1322
`CALibration:RESult?` on page 1323

4.7.2.2 Results

For each channel, the results of the individual alignment steps are shown for all technical channel component. In case you require support, you may be asked to provide this information.

Access: "Settings" >"Maintenance" > "Alignment" > "Show results"



Self alignment step	Alignment step results
THA offset	Init
THA gain	Init
Spc	Init
Deskew	Init
Deskew interleaved	Init
VarGain 50	Init
FixGain 50	Init
Offset 50	Init
BufFixGain	Init
BufVarGain 1M	Init
BufVarGain20dB1M	Init
FixGain 1M	Init
Offset 1M	Init
Frequency Response	Init
External Trigger	Init
Input Signal	Init

4.7.2.3 Performing a self-alignment

The self-alignment aligns the data from several input channels vertically and horizontally to synchronize the timebases, amplitudes and positions. The self-alignment process includes a basic hardware check.

Recommendation on performing the self-alignment:

- When putting the instrument into operation for the first time
- After a firmware update
- Once a week
- When major temperature changes occur (> 5°)

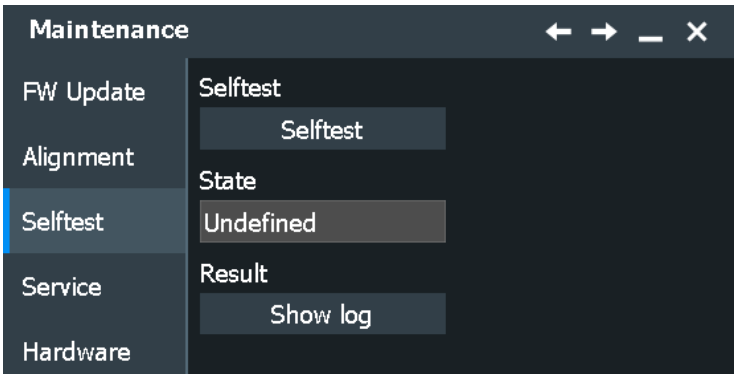
1. Warm up the instrument before you start the self-alignment. The minimum warm-up time is indicated in the data sheet.
2. Remove the probes from the input connectors.
3. In the "Menu" > "Settings" menu, select "Maintenance".
4. In the "Alignment" tab, tap "Start Alignment".

   The alignment is performed, the process might take several minutes. A message box informs you about the running process, wait until this message box closes. The overall pass/fail result is shown in the "Overall alignment state" field. The results of the individual alignment steps for each input channel are indicated in the "Results" tab. This information is required if problems arise.

4.7.3 Self-test

The instrument's self-test checks the hardware for correct operation. Perform the self-test if you suspect problems in hardware operation.

Access: "Settings" >"Maintenance" > "Selftest"



The test can take several minutes. The summary result is shown in the "State" field, which can be helpful in case you need support.

Selftest
Starts the self-test.

Remote command:
`*TST?` on page 1291

State
Shows the summary result of the self-test: Pass or Fail.

Remote command:
`DIAGnostic:SERVice:STST:STATe?` on page 1323

Result
Opens a log file with detailed information on the self-test steps and operation of hardware components, which can be helpful in case you need support.

4.7.4 Maintenance information

Access: "Settings" >"Maintenance" > "Hardware" tab > "Advanced"

The "Maintenance" dialog box provides information on your R&S RTO6 configuration, which can be helpful in case you need support.

System Info
This tab provides general information on the hardware configuration, and indicates where system information can be found on the instrument. Here you can also show the content of the device footprint file.

Remote command:
`DIAGnostic:SERVice:PARTnumber` on page 1324
`DIAGnostic:SERVice:SERialnumber?` on page 1324
`SYSTem:DFPRint` on page 1323

Mainboard
This tab provides information on the mainboard configuration in your instrument.

Frontend
This tab provides information on the frontend configuration in your instrument.

Frontpanel
This tab provides information on the front panel module installed in your instrument.

MSO
This tab is only relevant if the MSO option R&S RTO6-B1 is installed. The tab provides information on the MSO hardware module that is installed in your instrument.

AWG
This tab is only relevant if the waveform generator option R&S RTO6-B6 is installed. The tab provides information on the generator hardware module that is installed in your instrument.

PSC
This tab is only relevant if the pulse source option R&S RTO6-B7 is installed. The tab provides information on the pulse source hardware module that is installed in your instrument.

Service
This tab allows the service personnel to enter a password that activates further service functions.

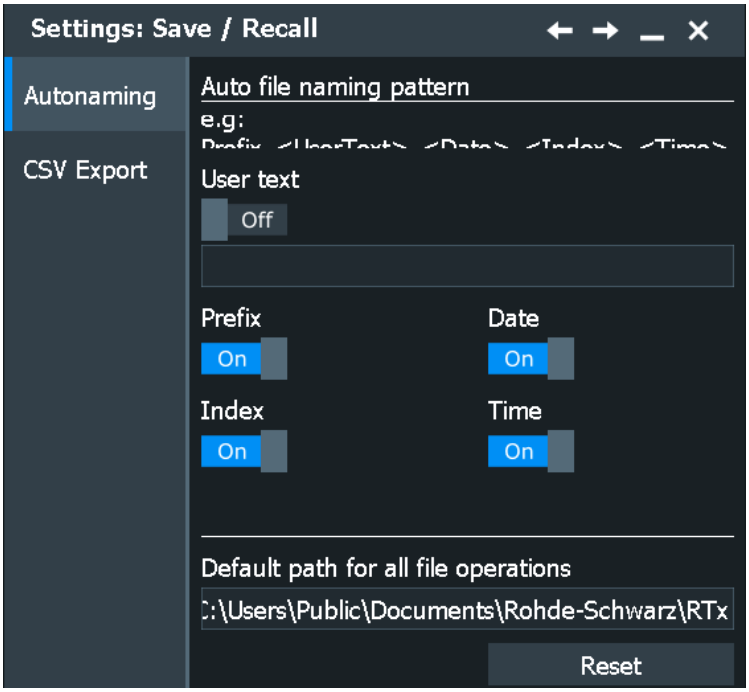
Remote command:
- `DIAGnostic:SERVice:PWD` on page 2572

4.8 Save / recall

4.8.1 Autonaming

Access:"Menu" > "Settings" key > "Maintenance" > "Autonaming" tab.

4.8.1.1 Autonaming settings



In this tab, you can define the pattern for automatic file name generation. This name is used as the default file name. The default path is the storage location for all saved files and their subdirectories.

Prefix
If enabled, inserts the default prefix in the file name. The prefix indicates the type of data that is saved, for example, Histogram, RefCurve, Settings.

Remote command:
`MMEMory:AUTonaming:PREFix` on page 1666

User text (enable)
If enabled, inserts the specified user text after the prefix.

Remote command:
`MMEMory:AUTonaming:USERtext` on page 1666

Text input
User-defined text to be inserted after the prefix.

Remote command:
`MMEMory:AUTonaming:TEXT` on page 1667

Date
If enabled, inserts the current date.

Remote command:
`MMEMory:AUTonaming:DATE` on page 1667

Index
If enabled, inserts an index.

Remote command:
`MMEMory:AUTonaming:INDex` on page 1667

Time
If enabled, inserts the current time.

Remote command:
`MMEMory:AUTonaming:TIME` on page 1667

Default path for all file operations
Defines the default path displayed in the file selection dialog box for loading and storing operations. If a USB flash drive is connected, the path is set automatically to the drive letter of the USB flash drive.

To switch the directory quickly, double-tap the input field. Use the symbols on the left of the file explorer box to change the directory.

Remote command:
`MMEMory:AUTonaming:DEFaultpath` on page 1667

Reset
Resets the default file path to the factory default.

Remote command:
`MMEMory:AUTonaming:RESPath` on page 1667
`MMEMory:AUTonaming:RESall` on page 1667

4.8.1.2 Defining default file paths and names

When a save or load operation is performed, a default file name and path is provided. You can configure which path is used and how the file name is generated. In the file selection dialog box, you can change the folder and name as desired.

To define the default file path

1. Tap "Menu" > "Settings".
2. Select the "Save /Recall" tab.
3. Select the "Autonaming" tab.
4. Double-tap the "Default path for all file operations" field.

   The directory selection dialog box is opened.
5. Select the folder in which the data is stored by default. Use the symbols on the left of the file explorer box to switch to often used directories.
6. To restore the factory-set default path, tap "Reset" next to the path field.

To define the automatic file name pattern

The automatic file name pattern can consist of the following elements:

<Prefix>_<UserText>_<Date>_<Index>_<Time>

The prefix depends on the data type to be stored and cannot be changed by the user. The other elements can be enabled or disabled as required.

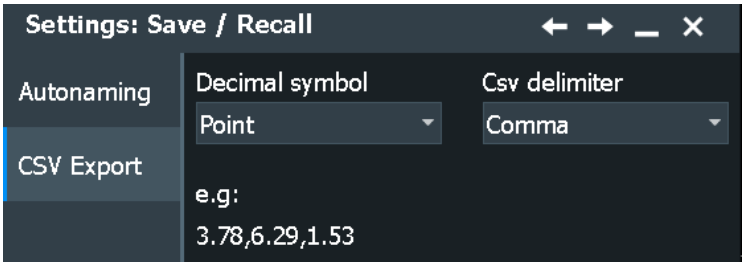
1. Tap "Menu" > "Settings".
2. Select the "Save /Recall" tab.
3. Select the "Autonaming" tab.
4. To insert a user-defined text after the prefix, enter the text in the edit field. and enable "User text".
5. If you want to exclude the prefix, current date, time or an index (serial number), disable the corresponding option.

   The specified elements are used to generate the default file name for the next storage operation.

4.8.2 CSV export

Access: "Menu" > "Settings" key > "Save /Recall" > "CSV Export" tab.

In this dialog, you can define the format of your CSV file.



Decimal symbol
Selects if point or comma is used as a decimal symbol in the exported CSV file.

Remote command:
`EXPort:RESult:DECSymbol` on page 1668

Csv delimiter
Selects the list separator symbol from a list. Available are semicolon, comma, space, tab and colon.

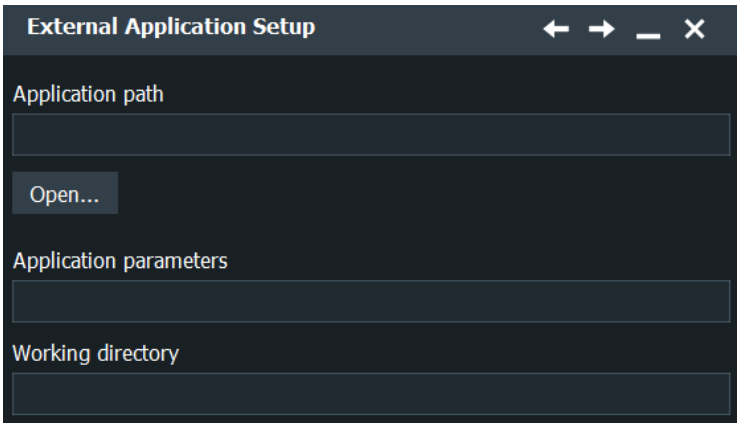
Remote command:
`EXPort:RESult:DELimiter` on page 1668

4.8.3 External application

The R&S RTO6 can start an external application on the instrument or in the network (if connected) when an event occurs.

The following events can start an application:

- Trigger event
- Mask violation
- Successful completion of mask test
- Limit or margin violation of measurements
- Successful completion of limit and margin tests



Set the path of the application executable, optional parameters, and the working directory as in a Windows shortcut definition. The setup is valid for all events.

Remote commands:

- `EXECutable:NAME` on page 1668
- `EXECutable:PARameter` on page 1669
- `EXECutable:WDIRectory` on page 1669

5 Acquisition and waveform setup

This chapter describes the horizontal and vertical settings as well as the acquisition and probe setup.

5.1 Basics

This chapter provides background information on the essential settings in the vertical and horizontal systems, on acquisition setup and probing.

5.1.1 Vertical system

The controls and parameters of the vertical system are used to scale and position the waveform vertically.

5.1.1.1 Input coupling

The input coupling influences the signal path between input connector and the following internal signal stage. The coupling can be set to DC, AC, or ground.

- DC coupling shows all parts of an input signal. DC coupling is available with 1 MΩ input impedance to connect standard passive probes. DC 50 Ω coupling is the default for 50 Ω input impedance to connect, for example, active probes.
- AC coupling is useful if the DC component of a signal is of no interest. AC coupling blocks the DC component of the signal so that the waveform is centered on zero volts.
- Ground coupling disconnects the input signal from the vertical system to see the ground level (zero volts) on the screen. Ground coupling is useful for reference purposes.

5.1.1.2 Vertical scale and position

Vertical scale and vertical position directly affect the resolution of the waveform amplitude. The vertical scale corresponds to the ADC input range. To get the full resolution of the ADC, set up the waveforms to cover most of the height of the diagram.

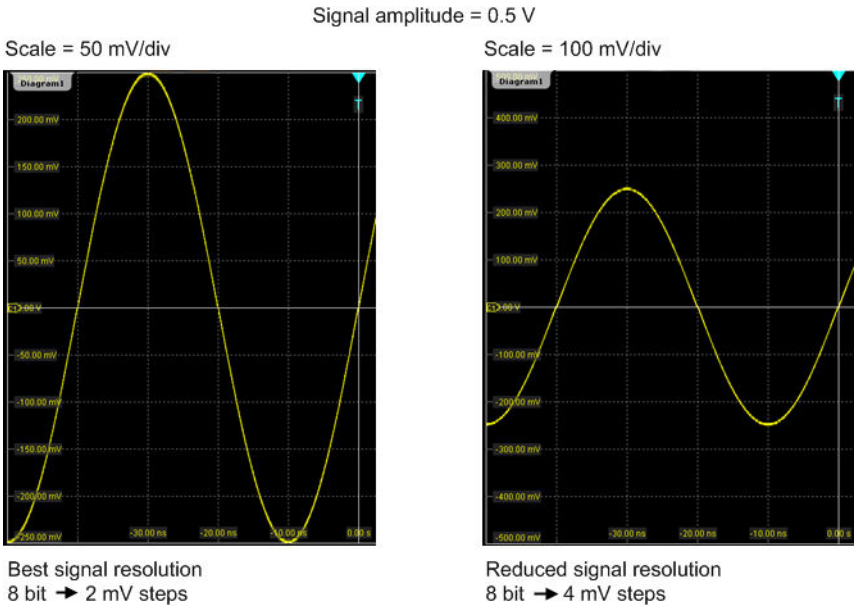


Figure 5-1: Input range and resolution of the ADC

With R&S RTO6, you can work with multiple diagrams, and each diagram obtains the full vertical resolution, no matter where the diagram is placed. Therefore, use a separate diagram for each waveform instead of the traditional setup that arranges the waveforms side by side in one diagram.

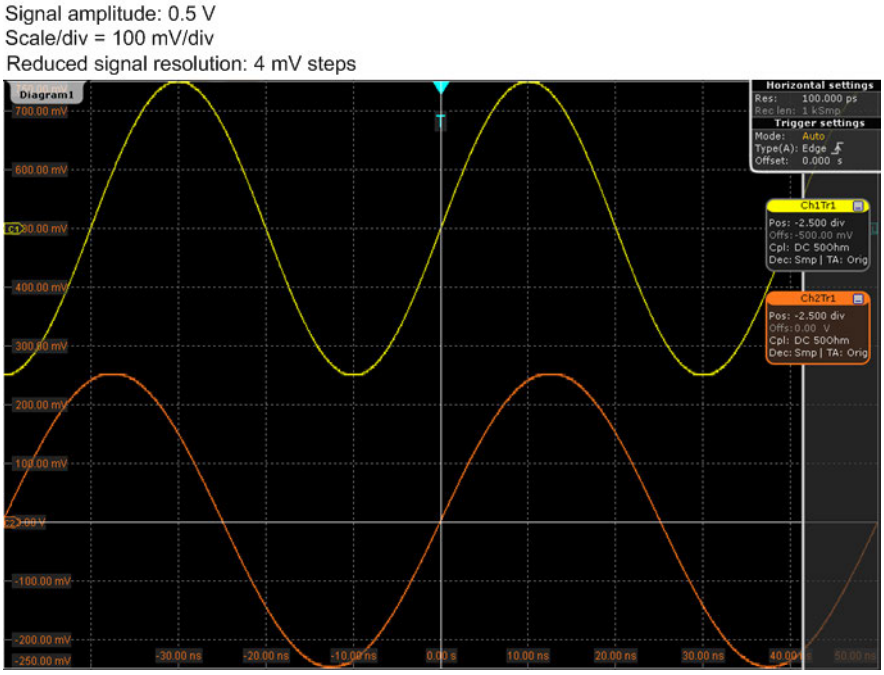


Figure 5-2: Traditional setup of multiple waveforms in one diagram: reduced resolution

Signal amplitude: 0.5 V
Scale = 50 mV/div
Best signal resolution: 2 mV steps

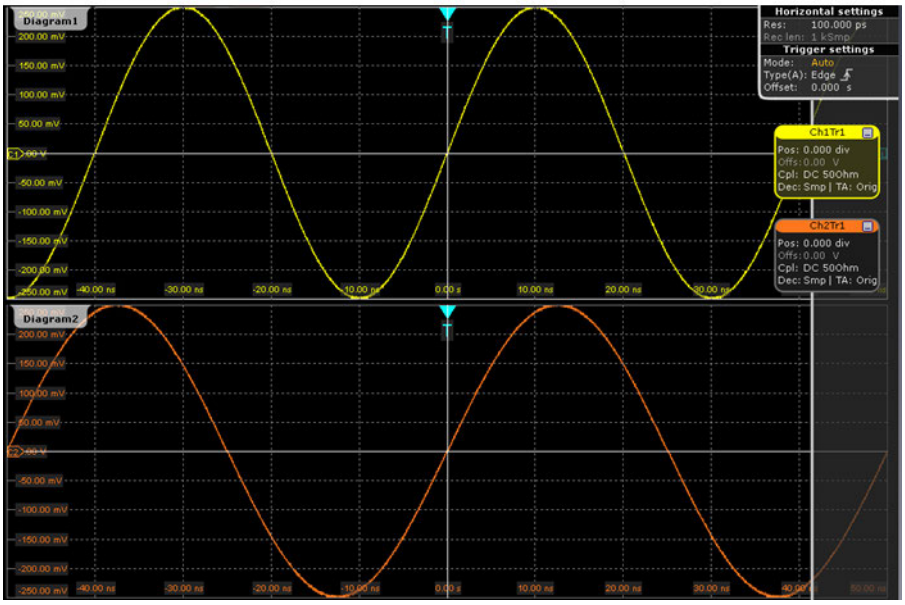


Figure 5-3: R&S RTO6 setup of multiple waveforms in separate diagrams: best resolution

5.1.1.3 Bandwidth

For analog applications, the highest signal frequency determines the required oscilloscope bandwidth. The oscilloscope bandwidth should be slightly higher than the maximum frequency included in the analog test signal to measure the amplitude with very little measurement error.

Most test signals are more complex than a simple sine wave and include several spectral components. A digital signal, for example, is built up of several odd harmonics. As a rule, for digital signals the oscilloscope bandwidth should be 5 times higher than the clock frequency to be measured.

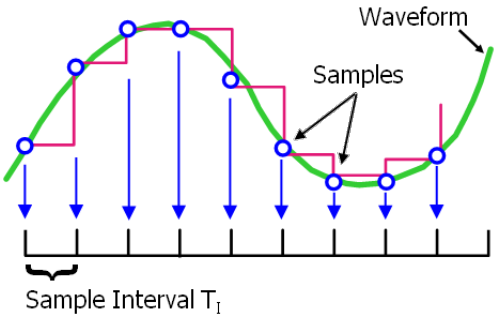
The oscilloscope is not a stand-alone system. You need a probe to measure the signal of interest, and the probe has a limited bandwidth, too. The combination of oscilloscope and probe creates a *system bandwidth*. To reduce the effect of the probe on the system bandwidth, the probe bandwidth must exceed the bandwidth of the oscilloscope, the recommended factor is 1.5 x oscilloscope bandwidth. See also: Chapter 5.1.4.1, "Voltage probes", on page 116

5.1.2 Sampling and acquisition

The vertical system of a digital oscilloscope conditions the test signal in a way that the following A/D converter (ADC) can transform the measured voltage into digital data.

5.1.2.1 Sampling and processing

The A/D converter samples the continuous signal under test at specific points in time and delivers digital values called **ADC samples**. The rate at which the converter is working is the **ADC sample rate**, a constant value specified in GHz: $f_{ADC} = 1 / T_I$



The digital ADC samples are processed according to the acquisition settings. The result is a waveform record that contains **waveform samples** and is stored in the **waveform memory**. The waveform samples are displayed on the screen and build up the waveform.

The number of waveform samples in one waveform record is called **record length**. The rate of recording waveform samples - the number of waveform samples per second - is the **sample rate**. The higher the sample rate, the better the resolution is and the more details of the waveform are visible.

A sufficient resolution is essential for correct reconstruction of the waveform. If the signal is undersampled, aliasing occurs - a false waveform is displayed. To avoid aliasing and accurately reconstruct a signal, Nyquist theorem postulates that the sample rate must be at least twice as fast as the highest frequency component of the signal. However, the theorem assumes ideal conditions, so the Nyquist sample rate is usually not sufficient.

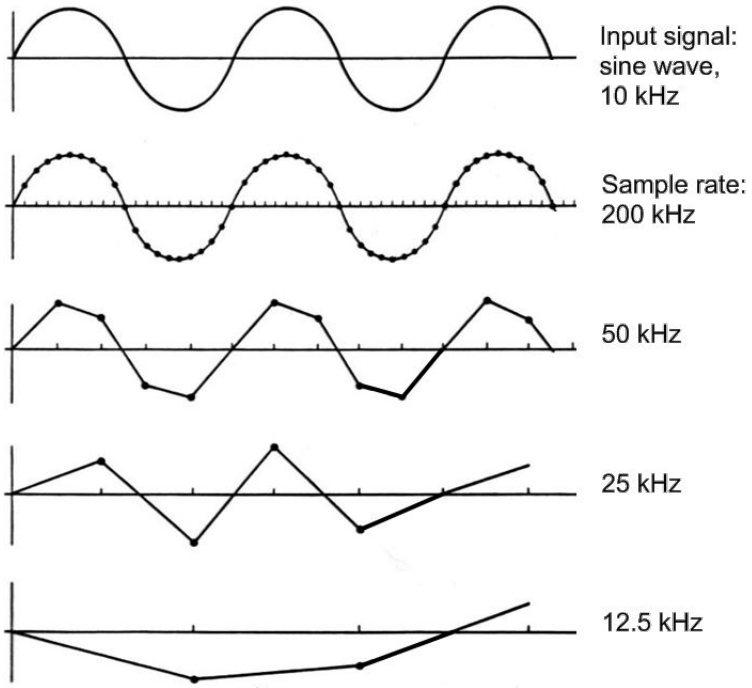


Figure 5-4: Waveforms acquired with different sample rates

To avoid aliasing, the sample rate must be set to a value 3 to 5 times the fastest frequency component of the signal. A higher sample rate increases signal fidelity, increases the chance to capture glitches and other signal anomalies, and improves the zoom-in capabilities.

5.1.2.2 Acquisition settings

The sample rate can be the same as the constant ADC sample rate, or higher, or lower. To get a higher sample rate, interpolation as method of **resolution enhancement** is used. To reduce the sample rate, **decimation** methods help: sample, peak detect, high resolution and RMS.

As digital waveform data is stored in the memory, and the memory can save many waveform records, further **waveform arithmetic** processing is possible: average and envelope waveforms are resulting waveforms, created from a composite of sample points taken from multiple acquisitions.

You can display up to three waveforms from one input signal and apply different decimation and arithmetic to each waveform.

5.1.2.3 Acquisition control

You can run the R&S RTO6 in two ways:

- Run Stop: the instrument acquires data until you stop it manually.
- Single: the instrument samples and processes a specified number of acquisitions.

The determining point of an acquisition is the trigger. The instrument acquires continuously and keeps the sample points to fill the pre-trigger part of the waveform record. When the trigger occurs, the instrument continues acquisition until the post-trigger part of the waveform record is filled. Then it stops acquiring and waits for the next trigger. When a trigger is recognized, the instrument does not accept another trigger until the acquisition is complete.

The trigger modes define how the instrument triggers:

- Normal: The instrument acquires a waveform only if a real trigger occurs, that is, if all trigger conditions are fulfilled.
- Auto: The instrument triggers repeatedly after a fixed time interval if the trigger conditions are not fulfilled. If a real trigger occurs, it takes precedence. If the real trigger is faster than the auto trigger, both modes are virtually the same.

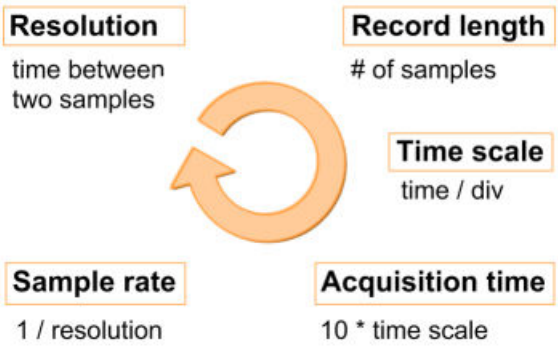
In practice, both trigger modes are useful: The auto mode lets you see the signal with little adjustment, while the normal mode selects the interesting part of the waveform. If you want to acquire a specified number of waveforms, make sure to select the normal trigger mode. Thus you get only the required number of interesting acquisitions.

See also: Chapter 6, "Triggers", on page 190.

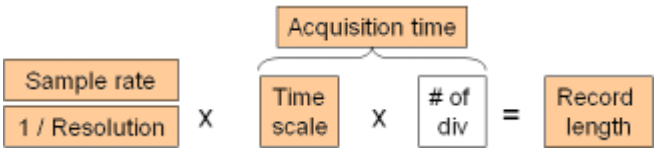
5.1.3 Horizontal system

5.1.3.1 Parameters of the horizontal system

The control parameters of the horizontal system are tightly connected. Thus, changing one parameter affects the other parameters as well.



The mathematical dependencies can be summarized as follows:



The number of divisions is 10, which is the only constant parameter.

When you set up horizontal parameters, you can choose whether the record length or the resolution remains constant.

- With constant resolution, increasing the time scale also increases the record length, and vice versa. You can limit the record length to a maximum value.
- With constant record length, increasing the time scale coarsens the resolution, that is, the time between two waveform samples gets longer.

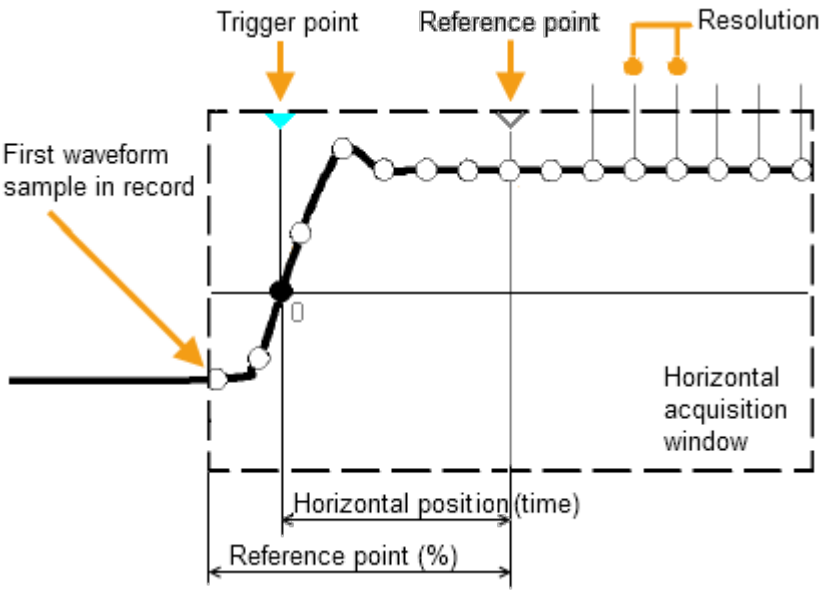
For both settings, the "Auto adjustment" ensures a sufficient resolution to prevent undersampling.

5.1.3.2 Horizontal position

As described before in Chapter 5.1.2.3, "Acquisition control", on page 113, the trigger is the determining point of the waveform record.

In many scenarios, you want to analyze the waveform some time before or after the trigger. To adjust the horizontal acquisition window to the waveform section of interest, you can use the following parameters:

- The **horizontal position** defines the time distance from the trigger point (the zero point of the diagram) to the reference point. Changing the horizontal position, you can move the trigger point, even outside the screen.
- The **reference point** is the rescaling center of the time scale on the screen. If you modify the time scale, the reference point remains fixed on the screen, and the scale is stretched or compressed to both sides of the reference point.



5.1.4 Probes

A probe connects the signal source (DUT) to the oscilloscope, and delivers the signal to be measured. It is the essential first link in the measurement chain.

An ideal probe fulfills the following requirements:

- Safe and reliable contacts
- Infinite bandwidth
- The probe should not load the signal source and thus impact the circuit operation.
- The connection should not introduce or suppress signal components (hum, noise, filter) and thus degrade or distort the transferred signal.

In reality, the probe can never be an ideal one, it always affects the signal transmission and the signal source, and thus the measured signal. It depends on the frequency to be measured and on the signal source to determine the acceptable loading, and to determine which kind of probe delivers good results.

The solution depends on the quantity to be measured regarding:

- Signal type: voltage, current, power, pressure, optical, etc.
- Signal amplitude: The oscilloscope itself can only display voltages in a limited range. Most probes can adjust the dynamic range to amplitudes from a few mV to 10 V. Smaller or much larger signals require specialized equipment.
- Signal frequency: High frequencies require advanced equipment to get correct results.
- Source characteristic: The source impedance is the decisive factor when choosing the suitable connection.

5.1.4.1 Voltage probes

The following table provides an overview on common voltage probes and their usage.

Table 5-1: Voltage probes overview

Probe type	Attenuation	Typical bandwidth range	Oscilloscope input	Usage
Passive, high impedance	1:1	10 MHz	1 MΩ	Low-speed signals, low-level signals
Passive, high impedance	10:1	500 MHz	1 MΩ	General purpose
Passive, low impedance	10:1	up to 10 GHz	50 Ω	High frequency
Active, single-ended	10:1	up to 10 GHz	50 Ω	High speed
Active, differential	10:1		50 Ω	Floating

For a list of recommended probes, refer to the R&S RTO6 product brochure.

Besides the possible input voltage range, two factors are important when selecting a voltage probe: Bandwidth and impedance over frequency.

- **Bandwidth:**
  The combination of probe and oscilloscope builds up a system. The resulting system bandwidth is approximately determined with:

$$\frac{1}{BW_{system}} = \sqrt{\left(\frac{1}{BW_{probe}}\right)^2 + \left(\frac{1}{BW_{scope}}\right)^2}$$

To measure the signal with low measurement error, the system bandwidth should be higher than the highest frequency component of the signal. The probe bandwidth must be even higher than the system bandwidth.

- **Impedance:**
  A minimum impedance is required to keep the circuit loading low. Over frequency, the impedance decreases, in particular with passive probes. The probe impedance should be approximately 10 times the impedance of the circuit test point at the highest signal frequency.

Passive voltage probes

Passive probes have the following qualities:

- No active components inside
- BNC connector for universal use
- Compensate the probe when it is connected to a scope input: LF compensation matches the probe (mainly cable) capacitance to the oscilloscope input capacitance.
- With high impedance probes, the impedance varies significantly over frequency.
- With low impedance probes, the impedance variation over frequency is low, but the load on the source is high.

If you use passive probes, remember some recommendations:

- Use a probe recommended for your oscilloscope model.
- Use a ground lead as short as possible to minimize the effect of ground lead inductance. The resonance frequency can be much lower than the system bandwidth and thus can affect the measurement results, in particular, if you measure steep edge rise times.
- Select a probe that has a bandwidth of 5 to 10 times the highest frequency being measured. This bandwidth preserves the harmonics and thus the waveform integrity.

Active voltage probes - general

Active probes require operating power from the instrument and have a proprietary interface to the instrument. Their main qualities are:

- Low loading on signal source
- The probe is automatically recognized by the instrument, no adjustment is required.
- Adjustable DC offset at probe tip allows for high resolution on small AC signals which are superimposed on DC levels.
- Connections should be as short as possible to keep the usable bandwidth high.
- Observe the operating voltage range.
- The probe impedance depends on the signal frequency.

RT-ZS single-ended active probes and RT-ZD differential active probes provide special features for easier use and precise measurements. These special features are not available on RT-ZSxxE probes.

- The micro button on the probe head remotely controls important functions on the instrument, like running and stopping the acquisition, autoset, AutoZero and setting the offset to mean value.
- The R&S ProbeMeter measures DC voltages between the probe tip and the ground connection with very high precision. The result is displayed on the instrument's screen. So you can check DC voltages with different levels without having to adjust the measurement range of the oscilloscope. The R&S ProbeMeter also measures the zero error of the probe to optimize measurement results at small signal levels.

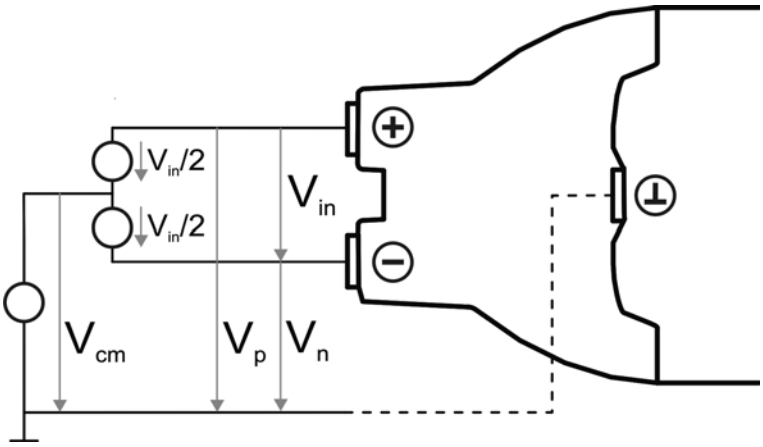
When you connect an R&S RT-ZSxx active probe to a channel input of the R&S RTO6, the oscilloscope recognizes the probe. It reads the identification and calibration data from the probe box and shows the result in the "Setup" and "Probe Attributes" tabs. This data together with the deskew time for a given channel is stored and processed by the R&S RTO6. If you connect the probe the next time to the same channel, the information is fetched and used.

Differential active probes

Differential active probes are designed to measure signals that are referenced against each other, and voltages that are not references to ground, for example twisted-pair signal lines. The R&S RT-ZD probes are differential probes with high input impedance, they can be used to measure voltages between any two test points.

Compared with two-channel measurement setup with single-ended probes, the measurement with differential probes is symmetric due to the same amplification and cable length on both paths. It is also immune to interference and noise and occupies only one input channel.

A differential probe has three sockets: the positive signal socket (+), the negative signal socket (-), and the ground socket.



Multiple input voltages can be defined for a differential probe:

- Differential mode input voltage (V_{in}, V_{dm})

Voltage between the positive and negative signal sockets

- Positive single-ended input voltage (V_p)
  Voltage between the positive signal socket and the ground socket
- Negative single-ended input voltage (V_n)
  Voltage between the negative signal socket and the ground socket
- Common mode input voltage (V_{cm})
  Mean voltage of positive and negative signal sockets referred to the ground socket, respectively

Two of these voltages are independent values, the other two can be calculated:

$$V_{in} = V_p - V_n$$

$$V_{cm} = \frac{V_p + V_n}{2}$$

R&S RT-ZD probes detect only differential input voltages and provide it to the oscilloscope. Common mode signals are suppressed by the probe. This characteristic is described by the common mode rejection ratio (CMRR):

$$CMRR = \frac{DifferentialGain}{CommonMode\,Gain}$$

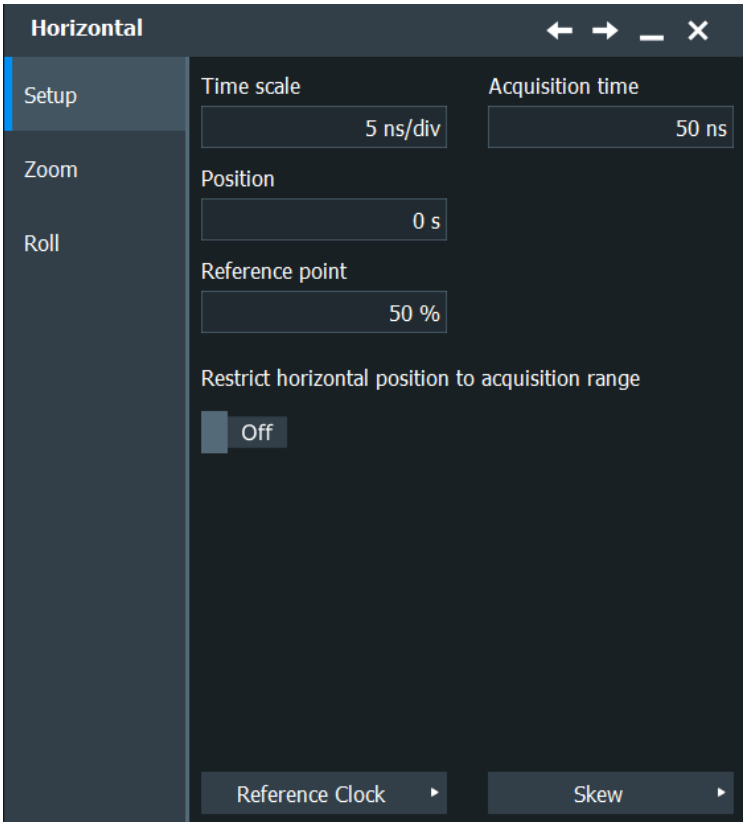
In addition, the R&S ProbeMeter of R&S RT-ZD differential probes can measure differential and common mode DC voltages. The measurement result is displayed on the oscilloscope's screen. The common mode measurement of the R&S ProbeMeter allows you to check the input voltage relative to ground. Thus, the CM measurement is a convenient way to detect breaches of the operating voltage window, and the reason of unwanted clippings.

5.2 Horizontal

The "Horizontal" dialog provides the time base configuration for channel and spectrum waveforms.

5.2.1 Setup settings

Access: "Menu" > "Horizontal" > "Setup" tab.



Time scale
Sets the horizontal scale for all channel and math waveforms in seconds per division. Increase the scale to see a longer time interval of the waveform. Decrease the scale to see it in more detail. The scale has a point that remains fixed on the screen when the scale value is changing - the reference point.

Remote command:
`TIMebase:SCALe` on page 1326

Acquisition time
Shows the time of one acquisition, that is the time across the 10 divisions of the diagram:

Acquisition time = Time scale * 10 divisions

Changing the acquisition time changes the time scale too.

Remote command:
`TIMebase:RANGe` on page 1327

Position
Defines the time distance between the reference point and the trigger point (the zero point of the diagram). If you want to see a section of the waveform some time before or after the trigger, enter this time as horizontal position. The requested waveform section is shown around the reference point. Use positive values to see waveform sections after the trigger - the waveform and the diagram origin move to the left.

See also "Reference point" on page 121.

Remote command:
`TIMebase:HORizontal:POSition` on page 1327

Reference point
Sets the position of the reference point in % of the screen. The reference point marks the rescaling center of the time scale. It is indicated by a grey triangle outline at the top of the diagram. If you modify the time scale, the reference point remains fixed on the screen, and the scale is stretched or compressed to both sides of the reference point.

Remote command:
`TIMebase:REFerence` on page 1327

Restrict horizontal position to acquisition range
If enabled, the horizontal position cannot be set outside the visible waveform diagram.

Remote command:
`TRIGger<m>:OFFSet:LIMited` on page 1328

Refence Clock
Opens a dialog for configuring the "Refence Clock" settings, see Chapter 5.3.2, "Reference clock (OCXO)", on page 125.

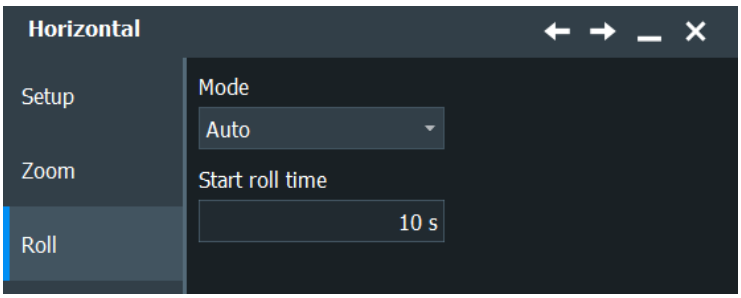
Skew
Opens a dialog for configuring the "Skew" settings, see Chapter 5.3.1, "Skew", on page 123.

5.2.2 Zoom settings

The zoom settings are described in Chapter 7.1, "Zoom", on page 239.

5.2.3 Roll settings

Access: "Menu" > "Horizontal" > "Roll" tab.



In this dialog, you can define the roll mode.

Roll mode
Selects, if the roll mode is set automatically by the instrument or if it is turned off.

In roll mode, the instrument shows the waveforms immediately, without waiting for the complete acquisition of the waveform record. If the time base is slow - at long time scale values - the roll mode saves waiting for the waveform display. The instrument displays newly acquired waveform points at the right edge of the display and moves the waveform to the left.

The roll mode has following restrictions:

- Roll mode disables persistence
- History is not available
- Event actions are not possible

The instrument activates the roll mode automatically if the following conditions are fulfilled:

- Acquisition time exceeds the defined "Minimum acquisition time"
- Waveform arithmetic is disabled ("Off")
- All channel waveforms are set to the same decimation mode, and only to one of these values: "Sample", "Peak detect", or "High res"
- All mask tests are disabled
- Fast segmentation is disabled
- Event actions are disabled
- FFT is disabled
- All serial buses are disabled
- All digital channels are disabled (MSO option R&S RTO6-B1)
- No CDR jitter data is acquired
- No zone trigger is active

The roll mode depends also on sample rate and record length. In roll mode, the sample rate limit is 20 MSample/s. At 50 s, the resulting record length limit is 1000 MSample.

If the acquisition time is >50 s, the record length limit is effective, and the maximum sample rate depends on the acquisition time:

Sample rate ≤ 1000 MSample / Acquisition time.

If the acquisition time is <50 s, the maximum sample rate in roll mode depends on the number of active channels:

Sample rate = 20 MSample/s / Number of active wfms.

The corresponding maximum record length is:

*Record length ≤ 20 MSample/s * Acquisition time / Number of active wfms*.

Thus, the roll mode switches off, or it does not activate automatically if:

- The record length exceeds the limit at acquisition times >50 s.
- The sample rate exceeds the limit.
- Too many waveforms are active.

Remote command:
`TIMebase:ROLL:ENABle` on page 1331
`TIMebase:ROLL:STATe?` on page 1331

Start roll time
The instrument can activate the roll mode automatically if the Acquisition time exceeds the value given here.

Remote command:
`TIMebase:ROLL:MTIMe` on page 1331

5.3 Horizontal accuracy

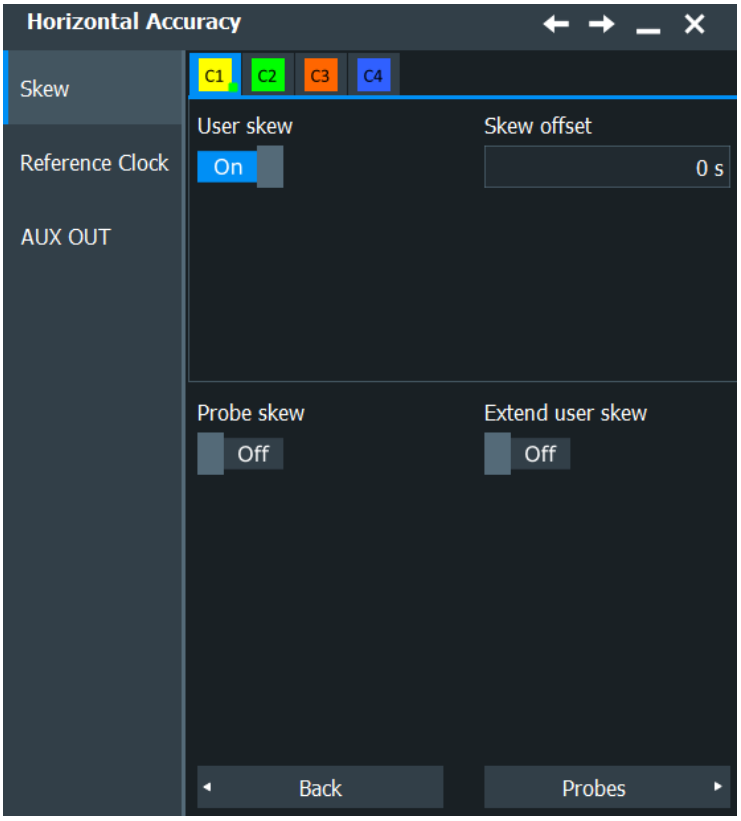
Access: "Menu" > "Horizontal" > "Setup" tab > "Skew".

The "Horizontal Accuracy" dialog box contains standard and optional settings to improve measurement and analysis accuracy and to reduce jitter effects.

5.3.1 Skew

Access: "Menu" > "Horizontal" > "Setup" tab > "Skew" > "Skew" tab.

Skew compensates signal propagation differences between channels caused by the different length of cables, probes, and other sources. Correct skew values are important for accurate triggering and timing relations between channels.



Make sure that the correct channel tab is selected.

User skew
If enabled, the "Skew offset" value is used for compensation.

Remote command:
`CHANnel<m>:SKEW:MANual` on page 1384

Skew offset
Sets a delay value, that is known from the circuit specifics but cannot be compensated by the instrument automatically. It affects only the selected input channel.

The offset range and possible values depend on Extend user skew.

Remote command:
`CHANnel<m>:SKEW:TIME` on page 1384

Probe skew
Measures the skew of all connected active probes and includes it in the total skew offset.

"Use probe group delay ..."
If enabled, the skew of all connected active probes is measured, displayed, and used for deskewing. The setting affects all active channels.

"Active probe"
Shows the type of the probe that is connected to the selected channel.

"Probe group delay"
Shows the result of the probe skew measurement on the selected channel.

Remote command:
`PROBe<m>:SKEState` on page 1385

Total skew offset
If Extend user skew is disabled, the sum of the measured "Probe group delay" and the "Skew offset" is shown. If "Use skew offset" is disabled, the skew offset is ignored.

If Extend user skew is enabled, only the probe skew is shown, and the skew offset is always ignored.

Extend user skew
Allows you to set higher value in "Skew offset" to compensate for the delay of the measurement setup. Without extension, the deskew range is ±100 ns, and delays shorter than the sample interval can be compensated.

With extension, the maximum delay is ±1 s, and the instrument can compensate complete samples. Compensation takes effect after the trigger. The maximum number of acquisitions in the memory is reduced.

Use the extended skew range to compare signals: One signal goes directly from the generator to the scope, and the other signal goes from the generator through the DUT to the scope. The delay of the DUT is higher than ±100 ns.

Remote command:
`CHANnel<m>:SKEW:EXTended` on page 1385

5.3.2 Reference clock (OCXO)

Access: "Menu" > "Horizontal" > "Setup" tab > "Skew" > "Reference clock" tab.

The oven-controlled crystal oscillator (OCXO) produces a 10 MHz internal reference signal with precise and stable frequency. With OCXO, you can also use an external reference signal. The input and output connectors for the external reference signal are on the rear panel alongside the external trigger input.

Detected
Indicates if the OCXO is detected by the instrument.

Oven hot
Indicates when the oven has reached its nominal temperature and is operating with the specified accuracy.

External reference
Sets the frequency of an external reference input signal that is connected to the external reference input on the rear panel of R&S RTO6. A frequency range from 1 MHz to 20 MHz is supported.

Remote command:
`SENSe[:ROSCillator]:EXTernal:FREQuency` on page 1388

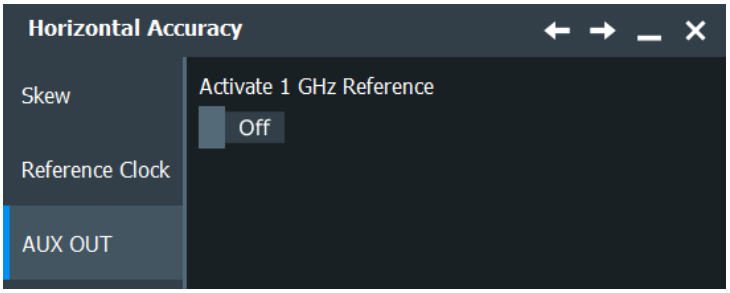
Use external reference
Enables the use of the external reference signal instead of the internal OCXO reference.

If an external reference is used, the frequency of the reference output signal is the same as of the reference input signal. Otherwise, the frequency of the reference output signal is 10 MHz, that is the frequency of the OCXO.

Remote command:
`SENSe[:ROSCillator]:SOURce` on page 1388

5.3.3 AUX OUT

Access: "Menu" > "Horizontal" > "Setup" tab > "Skew" > "AUX OUT" tab.



Activate 1GHz Reference
Enables the 1 GHz reference signal and sends it to the [Aux Out] connector. The signal is required for performance test to measure the frequency of the internal calibration signal.

Remote command:
`CALibration:SOURce:FREQuency` on page 1386
`CALibration:SOURce:STATe` on page 1386

5.4 Acquire settings

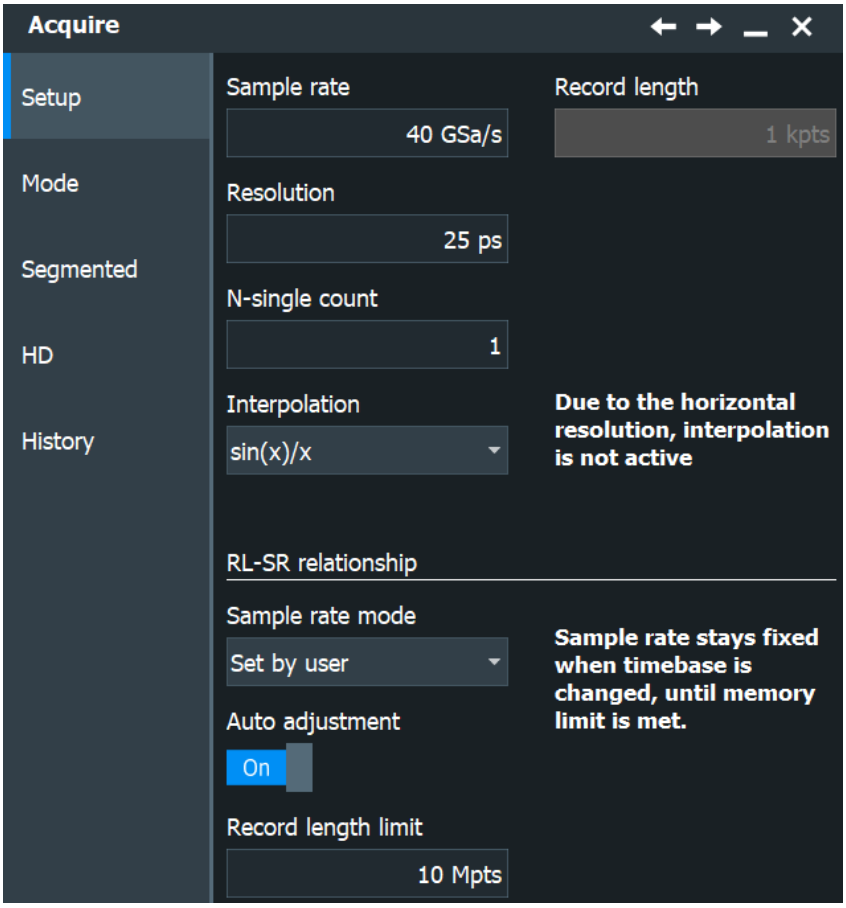
The "Acquire" dialog provides the acquisition configuration for channel and spectrum waveforms.

5.4.1 Setup settings

The "Setup" tab in the "Acquire" dialog provides the settings for the time axis.

For background information, see Chapter 5.1.3, "Horizontal system", on page 114.

Access: "Menu" > "Acquire" > "Setup" tab.



Sample rate
Sets the number of captured waveform points per second. It considers the samples of the ADC, and the reduction of waveform points by decimation.

If interpolation is not active, the sample rate is the reciprocal value of the resolution and thus also depends on the acquisition time and the record length.

If interpolation is active, the sample rate is limited to the ADC sample rate.

See also:
- Chapter 5.1.2, "Sampling and acquisition", on page 111
- Chapter 5.1.3, "Horizontal system", on page 114

Remote command:
`ACQuire:SRReal` on page 1330

Record length
Indicates the number of waveform samples that build the waveform across the acquisition time.

Remote command:
`ACQuire:POINts[:VALue]` on page 1330

Resolution
Sets the time between two waveform samples. A fine resolution with low values produces a more precise waveform record.

Remote command:
`ACQuire:RESolution` on page 1330

Average count (N-single count)
Access:
- "Menu" > "Acquire" > "Setup" tab > "N-single count"
- "Menu" > "Acquire" > "Mode" tab > "Average count"
- "Menu" > "Acquire" > "Segmented" tab > disable "Acquire maximum" > "Required"
- [Math] > "Arithmetic" tab > "Mode" is not "Off" > "Average count"

The acquisition and average count has several effects:
- It sets the number of waveforms acquired with [Single]
- It defines the number of waveforms used to calculate the average waveform.
  Thus, the instrument acquires sufficient waveforms to calculate the correct average if "Average" is enabled for waveform arithmetic. The higher the value is, the better the noise is reduced.
- It sets the number of acquisitions to be acquired in a fast segmentation acquisition series. Thus, you can acquire exactly one fast segmentation acquisition series with [Single].
  If fast segmentation is enabled and configured to acquire the maximum number of acquisitions, the acquisition count is set to that maximum number and cannot be changed.
  See also "Number of acquisitions" on page 132.
- It is the "Finished" criteria for the state of a mask test.

Remote command:
`ACQuire:COUNt` on page 1334

Interpolation
Selects the interpolation method. If the defined "Sample rate" is higher than the ADC sample rate, interpolation adds points between the captured samples of the waveform by various mathematic methods.

"Linear" Two adjacent ADC sample points are connected by a straight line, the interpolated points are located on the line. You see a polygonal waveform similar to the real signal, and also the ADC sample points as vertexes.

"sin (x)/x" Two adjacent ADC sample points are connected by a sin(x)/x curve, and also the adjoining sample points are considered by this curve. The interpolated points are located on the resulting curve. This interpolation method is precise and shows the best signal curve.

"Sample/Hold" The ADC sample points are displayed like a histogram. For each sample interval, the voltage is taken from the sample point and considered as constant, and the intervals are connected with vertical lines. Thus, you see the discrete values of the ADC - the measured samples.

Remote command:
`ACQuire:INTerpolate` on page 1332

Sample rate mode
You can choose to keep constant either the sample rate or the record length when you adjust the time scale or acquisition time.

See also: Chapter 5.1.3, "Horizontal system", on page 114

"Set by user" With constant sample rate and resolution, increasing the time scale also increases the record length, and vice versa. You can limit the record length to a maximum value.

"Fixed record length" With constant record length, increasing the time scale reduces the sample rate and coarsens the resolution, that is, the time between two waveform samples gets longer.

Remote command:
`ACQuire:POINts:AUTO` on page 1328

Auto adjustment
Prevents undersampling and ensures a sufficient resolution to acquire the correct waveform if the time scale is changed. The setting takes effect if the changed parameter - resolution or record length - reaches a limit. The instrument automatically keeps this parameter constant at its limit, and changes the other parameter regardless of the "Resolution / Record length" setting.

See also: Record length limit

Remote command:
`ACQuire:POINts:AADJust` on page 1329

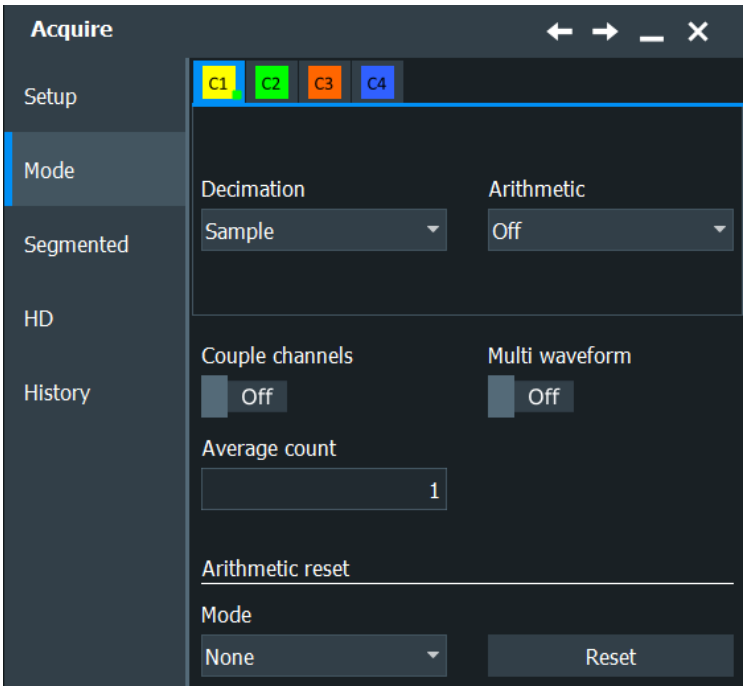
Record length limit
Sets a limit for the record length to prevent very large records. This value is only available if "Auto adjustment" is on and a constant resolution is selected. If you increase the time scale, the resolution remains constant and the record length increases until the limit is reached. Further increase of the time scale changes the resolution and keeps the record length limit.

See also:
- Record length
- Auto adjustment

Remote command:
`ACQuire:POINts:MAXimum` on page 1329

5.4.2 Mode settings

Access: "Menu" > "Acquire" > "Mode" tab.



Decimation
Selects the decimation mode. Decimation reduces the data stream of the ADC to a stream of waveform points with lower sample rate and a less precise time resolution. The R&S RTO6 uses decimation, if the waveform "Sample rate" is less than the ADC sample rate. In this case, interpolation is not possible.

There are different methods to define the recorded waveform point out of n sample points:

"Sample" One of n samples in a sample interval of the ADC is recorded as waveform point, the other samples are discarded. The time between the two adjacent waveform points is exactly the resolution. Very short glitches might remain undiscovered by this method.

"Peak detect" The minimum and the maximum of n samples in a sample interval are recorded as waveform points, the other samples are discarded.

"High res" The average of n sample points is recorded as one waveform sample. Averaging reduces the noise, the result is a more precise waveform with higher vertical resolution.

"RMS" The waveform point is the root mean square of n sample values. Thus, the RMS value reflects the instantaneous power. This arithmetic mode is used to average a measured power waveform. Linear averaging of power signals causes an error dependent on the noise of the signal to be averaged.

Remote command:
`CHANnel<m>[:WAVeform<n>]:TYPE` on page 1333

Arithmetic
Waveform arithmetic builds the resulting waveform from several consecutive acquisitions of the signal. The arithmetic works with interpolated and decimated waveforms.

This setting is waveform-specific.

The methods are:

"Off" The data of only one acquisition is recorded according to the decimation settings. In effect, no waveform arithmetic is processed.

"Envelope" Detects the minimum and maximum values in a sample interval over several acquisitions. Each acquisition is done in the "Peak detect" decimation mode, and the most extreme values for all acquisitions build the envelope. The resulting diagram shows two envelope waveforms: the minimums (floor) and maximums (roof).
The envelope is built until the restart criterion is reached, see "Arithmetic Reset > Mode" on page 131.

Note: If you change from "Envelope" to "Off", make sure to set also the "Decimation" to the required value.

"Average" The average is calculated from the data of the current acquisition and several acquisitions before. The method reduces random noise and other heterodyne signals. It requires a stable, triggered and periodic signal for correct function.
The number of acquisitions for average calculation is defined with "Average count"
The "Auto reset mode" defines the restart condition.

Remote command:
`CHANnel<m>[:WAVeform<n>]:ARIThmetics` on page 1334

Couple channels
Sets the acquisition mode and the waveform arithmetic of all channels to the last set value.

If the acquisition settings are coupled, "Multi waveform" is not available, only one waveform per channel can be used.

Remote command:
`ACQuire:CDTA` on page 1332

Multi waveform

For each channel, up to three waveforms can be shown and analyzed. The decimation mode and the waveform arithmetic are specific for each waveform. So you can analyze several aspects of the signal: For example, waveform1 shows the peaks, and waveform2 shows the average of the signal.

Remote command:
`ACQuire:MUWaveform` on page 1333

Average count (N-single count)

Access:
- "Menu" > "Acquire" > "Setup" tab > "N-single count"
- "Menu" > "Acquire" > "Mode" tab > "Average count"
- "Menu" > "Acquire" > "Segmented" tab > disable "Acquire maximum" > "Required"
- [Math] > "Arithmetic" tab > "Mode" is not "Off" > "Average count"

The acquisition and average count has several effects:
- It sets the number of waveforms acquired with [Single]
- It defines the number of waveforms used to calculate the average waveform. Thus, the instrument acquires sufficient waveforms to calculate the correct average if "Average" is enabled for waveform arithmetic. The higher the value is, the better the noise is reduced.
- It sets the number of acquisitions to be acquired in a fast segmentation acquisition series. Thus, you can acquire exactly one fast segmentation acquisition series with [Single].
  If fast segmentation is enabled and configured to acquire the maximum number of acquisitions, the acquisition count is set to that maximum number and cannot be changed.
  See also "Number of acquisitions" on page 132.
- It is the "Finished" criteria for the state of a mask test.

Remote command:
`ACQuire:COUNt` on page 1334

Arithmetic Reset > Mode

Defines when the envelope and average evaluation restarts.

"None"	No restart, the number of acquisitions considered by the waveform arithmetics is not limited.
"Time"	Restarts the envelope and average calculation after the time defined in "Time".
"Waveforms"	Restarts the envelope and average calculation after a number of acquired waveforms defined in "Count".

Remote command:
`ACQuire:ARESet:MODE` on page 1335
`ACQuire:ARESet:TIME` on page 1335
`ACQuire:ARESet:COUNt` on page 1335

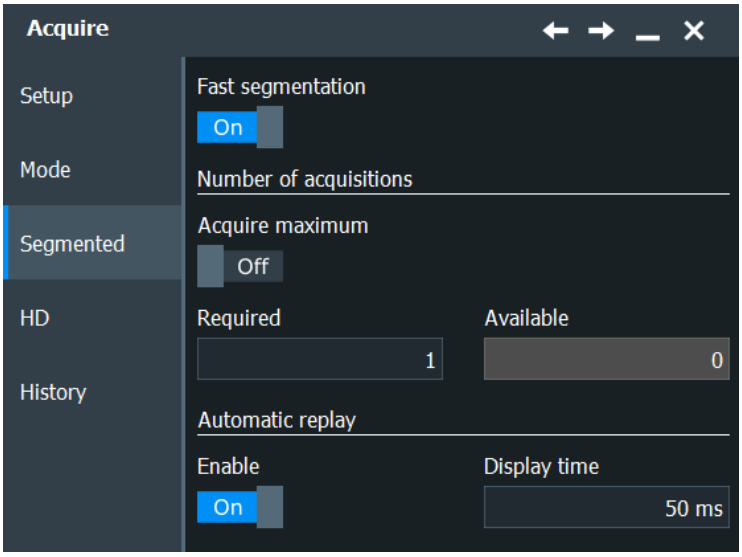
Reset

Forces the immediate restart of the envelope and average calculation for all waveforms.

Remote command:
`ACQuire:ARESet:IMMediate` on page 1335

5.4.3 Segmented settings

Access: "Menu" > "Acquire" > "Segmented" tab.



Fast segmentation
Switches the fast segmentation mode on and off.

Remote command:
`ACQuire:SEGMented:STATe` on page 1336

Number of acquisitions
You can define the number of acquisitions to be stored in a fast segmentation acquisition series:

- Acquire the maximum number of acquisitions that can be stored in the sample memory.
  To acquire the maximum number, enable "Acquire maximum". The maximum number of acquisitions is shown in the "Required" field.
- Acquire a given number of acquisitions.
  Enter the number in the "Required" field.

The acquisition count (Average count (N-single count)) is always set to the required number of acquisitions. Thus you can acquire exactly one fast segmentation acquisition series with [Single]. The [Run Stop] key works in the same way as [Single], it stops acquisition when the series is completed.

You can stop the running acquisition before the series is completed.

The number of acquired waveforms is shown in "Available" and can be displayed with "Show history".

Remote command:
`ACQuire:SEGMented:MAX` on page 1336

Enable automatic replay
If enabled, the instrument starts processing and displaying the data when the acquisition series is captured completely. Depending on the number of acquisitions, it can take some time until the acquisition series is displayed. If the setting is disabled, the instrument only captures the data and stores it in the sample memory.

Remote command:
`ACQuire:SEGMented:AUToreplay` on page 1336

Display time
Defines the display speed of the fast segmentation acquisition series. Display starts after the series has been captured completely.

5.4.4 High definition mode

The high definition mode offers up to 16 bits of vertical resolution. Higher vertical resolution reduces quantization noise and acquires waveforms of higher accuracy with finer details of the signal to be seen.

The number of vertical resolution bits defines the number of vertical levels that the acquisition samples are mapped to (quantization). 16 bits of resolution represent 65536 voltage quantization levels, while 8 bits of resolution represent only 256 voltage levels. The waveform values are recorded with 16 bit word length, except for peak detect decimation.

The higher vertical resolution is achieved by applying a digital low pass filter (DSP filter) to the output of the ADC, which reduces the bandwidth of the signal. Increasing the bandwidth reduces the resulting digital resolution. The high definition is also applied to the digital trigger, thus the R&S RTO6 can trigger with the same high resolution with which they can display signals.

High definition can be used, for example, to measure slow pulses with high accuracy, or to analyze AM signals with very low modulation index, as used in radar.

See also:

- Chapter 5.1.1, "Vertical system", on page 109
- Chapter 5.1.2, "Sampling and acquisition", on page 111

5.4.4.1 High definition settings

Access: [HD]

High definition is a special acquisition mode of the oscilloscope. This mode has only one setting - the filter bandwidth.

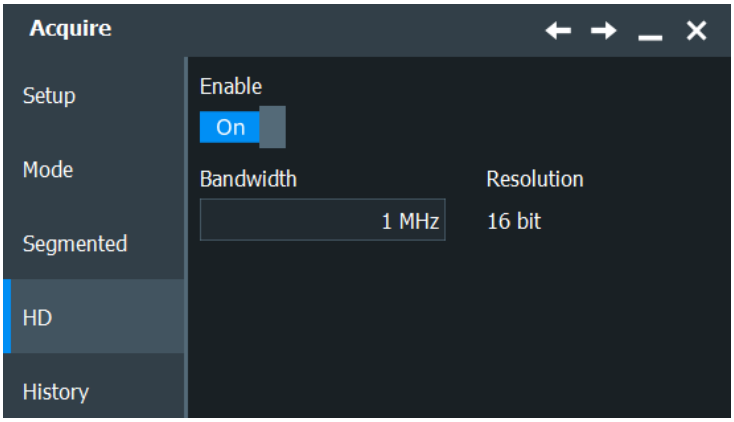


Figure 5-5: Setting the instrument into high definition mode

Enable
Enables the higher digital resolution, up to 16 bit.

Remote command:
`HDEFinition:STATe` on page 1387

Bandwidth
Sets the filter bandwidth for the high definition mode.

The maximum filter bandwidth depends on the instrument bandwidth.

Remote command:
`HDEFinition:BWIDth` on page 1387

Resolution
Shows the resulting vertical resolution in high definition mode. The higher the filter bandwidth, the lower the resolution. For details, refer to the R&S RTO6 Specifications.

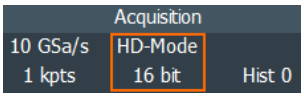
Remote command:
`HDEFinition:RESolution?` on page 1387

5.4.4.2 Effects of the high definition mode

The high definition mode has several effects:

Acquisition

The active high definition mode is indicated by "HD" in the "Acquisition" label.



The high definition mode works with half the realtime sample rate. For FFT, the instrument halves this sample rate again.

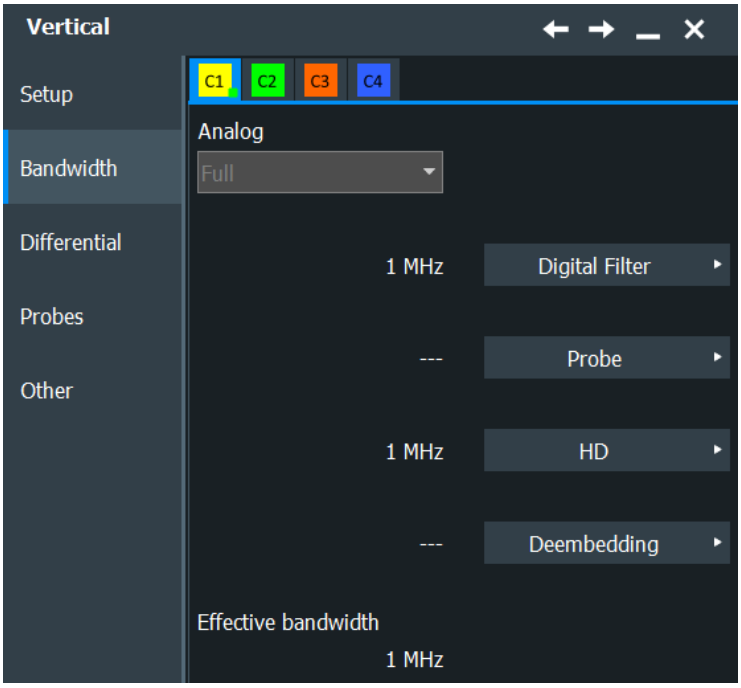
The waveform values are recorded with 16-bit word length, except for peak detect decimation (2 values with 8 bit).

Vertical system

The current bandwidth is shown in the channel label.

In the "Vertical" > "Bandwidth" dialog (C<x>), the "Bandwidth" setting is not available because the bandwidth is set by the high definition filter.

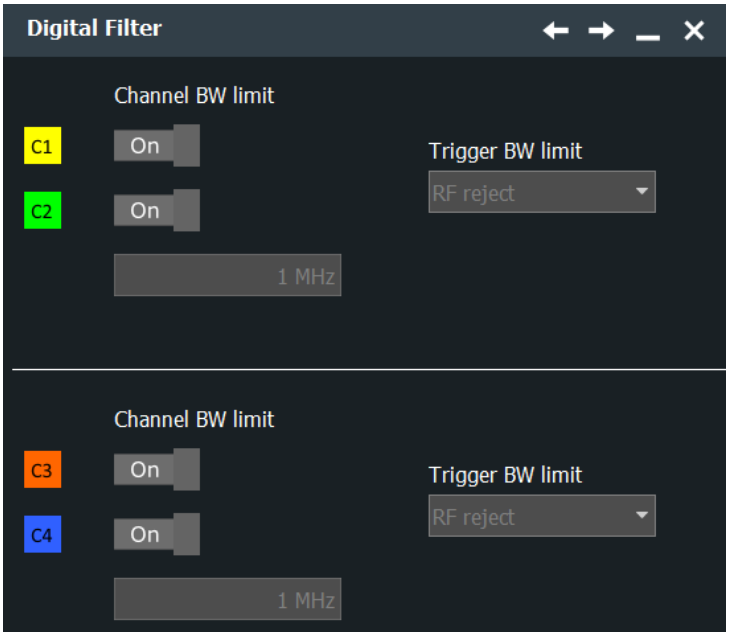
The minimum vertical scale is 500 µV/div instead of 1 mV/div in normal mode.



Digital filter

The digital filter settings are set automatically.

You can change the high definition "Bandwidth" in the "Acquire" > "HD" dialog.



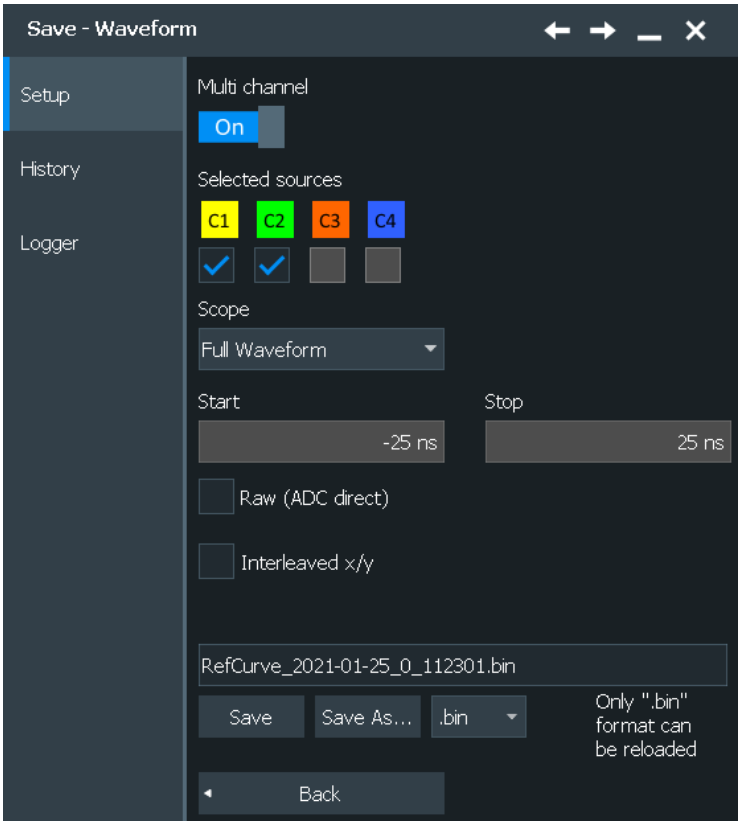
History

Due to the 16-bit word length, the history depth is reduced, less waveforms are saved than in normal mode.

Export

In high definition mode, waveform data in raw format is exported to file with 16-bit word length, except for peak detect decimation (2 values with 8 bit). In addition, you can define the byte order of the data words.

You can define additional export settings in the "Menu" key >"Save/Recall" key > "Save" tab > "Waveform" > "Setup" tab.



See:

- "Raw (ADC direct)" on page 461
- "Interleaved x/y" on page 462
- "Byte order" on page 1244

If you use remote control commands to transfer data to a controlling computer, set the data format to `INT,16` to transfer the complete data words (see `FORMat[:DATA]` on page 1292).

Further restrictions

The IQ mode (option R&S RTO6-K11) is not available if high definition mode is active.

5.4.5 History settings

The history settings are described in Chapter 7.4.2, "History setup", on page 276.

5.5 Vertical setup

The "Vertical" dialog box contains all channel-dependent settings and information.

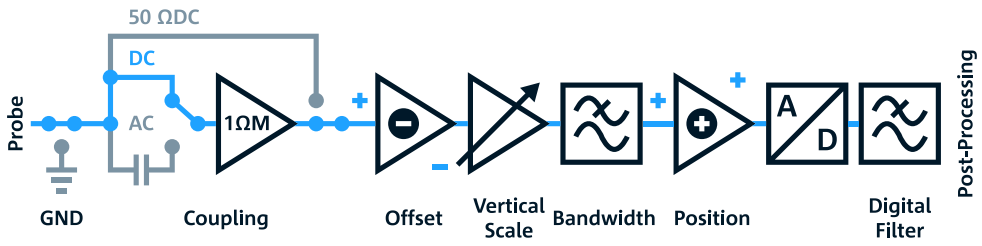
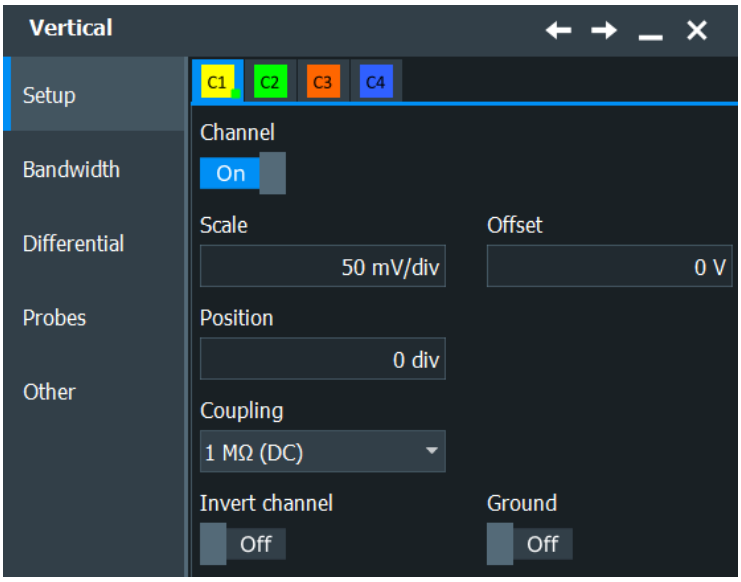


Figure 5-6: Signal path in the vertical system of R&S RTO6

5.5.1 Setup settings

Access: "Menu" > "Vertical" > "Setup"

The "Setup" tab provides all basic vertical settings. The channels are listed in horizontal subtabs. Make sure to select the correct channel tab before you enter the settings.



Channel On/Off 139
Scale 139
Offset 139
Position 140
Coupling 140
Invert channel 140
Ground 141

Channel On/Off
Switches the channel signal on or off. The signal icon appears on the signal bar. The waveform of the last acquisition is displayed in the diagram.

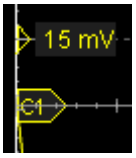
Remote command:
`CHANnel<m>:STATe` on page 1337

Scale
Defines the vertical scale in Volts per division. Increasing the scale compresses the display of the signal.

Remote command:
`CHANnel<m>:SCALe` on page 1338

Offset
The offset voltage is subtracted to correct a signal with DC offset. The vertical center of the selected channel is shifted by the offset value and the signal is repositioned within the diagram area. Negative offset values move up the waveform, positive values move it down.

The offset of a signal is determined and set by the autoset procedure. The current value is shown in the waveform label, and it is marked by a small triangle in the grid.



If a Rohde & Schwarz differential probe is connected, the offset is the differential offset.

If a Rohde & Schwarz modular probe is connected, the offset of the selected probe mode is used. For example, in CM mode, the offset is the common mode offset.

By default, the horizontal grid axis remains in the center when the offset is changed. To shift the axis together with the waveform, disable Keep Y-grid fixed in ""Settings" > Appearance > Grid".

Remote command:
`CHANnel<m>:OFFSet` on page 1339

Position
Moves the selected signal up or down in the diagram. The visual effect is the same as for Offset but the waveform is adjusted later in the signal flow. While the offset sets a voltage, position is a graphical setting given in divisions.

By default, the horizontal grid axis remains in the center when the offset is changed. To shift the axis together with the waveform, disable Keep Y-grid fixed in ""Settings" > Appearance > Grid".

Remote command:
`CHANnel<m>:POSition` on page 1339

Coupling
Selects the connection of the channel signal. The coupling determines what part of the signal is used for waveform analysis and triggering.

"DC 50 Ω"	Connection with 50 Ω termination, passes both DC and AC components of the signal.
"DC 1 MΩ"	Connection with 1 MΩ termination, passes both DC and AC components of the signal.
"AC"	Connection with 1 MΩ termination through DC capacitor, removes DC and very low-frequency components. If AC coupling is set, the attenuation of passive probes has no effect, and voltage is applied to the instrument with factor 1:1. Observe the voltage limits, otherwise you can damage the instrument.

Remote command:
`CHANnel<m>:COUPling` on page 1337

Invert channel
Turns the inversion of the signal amplitude on or off. To invert means to reflect the voltage values of all signal components against the ground level. If the inverted channel is the trigger source, the instrument triggers on the inverted signal.

You can use inversion, for example, to switch the polarity of a differential signal without changing the probe connections.

Remote command:
`CHANnel<m>:INVert` on page 1339

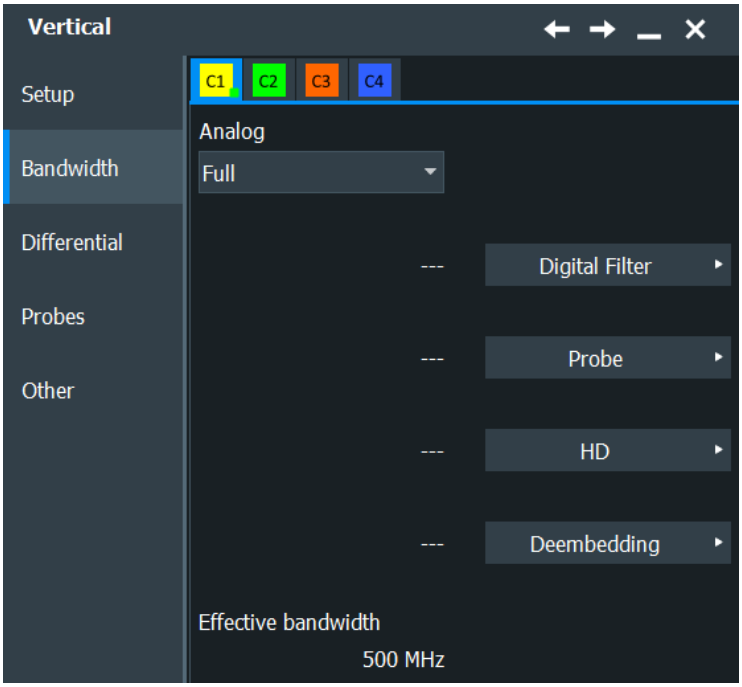
Ground
Connects the input to the ground.

Remote command:
`CHANnel<m>:GND` on page 1338

5.5.2 Bandwidth settings

Access: "Menu" > "Vertical" > "Bandwidth"

The "Bandwidth" tab provides all settings that affect the bandwidth of the measurement system. The channels are listed in horizontal subtabs. Make sure to select the correct channel tab before you enter the settings.



Analog
Selects the bandwidth limit.

The specified bandwidth indicates the range of frequencies that the instrument can acquire and display accurately with less than 3dB attenuation. The probe has also a limited bandwidth and thus affects the resulting system bandwidth.

See also: Chapter 5.1.1.3, "Bandwidth", on page 111

"Full" At full bandwidth, all frequencies in the specified range are acquired and displayed. Full bandwidth is used for most applications.

"20 MHz, 200 MHz, 800 MHz"
Frequencies above the selected limit are removed to reduce noise at different levels.

Remote command:
`CHANnel<m>:BANDwidth` on page 1340

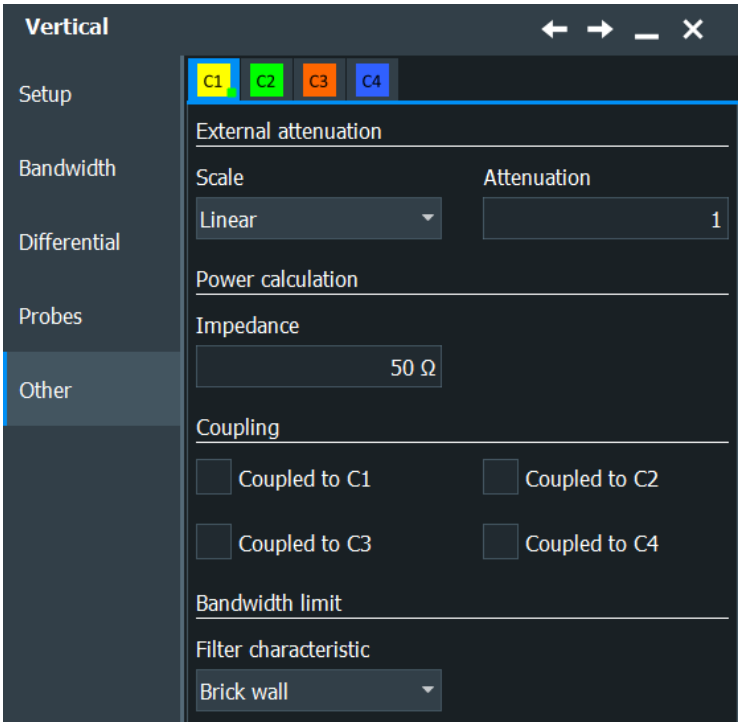
Effective bandwidth
Displays the effective bandwidth.

5.5.3 Probes settings

See Chapter 5.6, "Probes", on page 145.

5.5.4 Other settings

Access: "Menu" > "Vertical" > "Other"



External attenuation: Scale, Attenuation
Consider a voltage divider that is part of the DUT before the measuring point. The external attenuation is included in the measurement, and the instrument shows the results that would be measured before the divider. External attenuation can be used with all probes.

"Scale" Select linear or logarithmic attenuation scale.

"Attenuation" Enter the attenuation of the voltage divider according to the selected scale. The conversion from linear to logarithmic values depends on the "Vertical unit" of the probe:
For voltage-based unit (V and A):
*attenuation (dB) = 20 * log_{10}(attenuation factor)*
For power-based unit (W):
*attenuation (dB) = 10 * log_{10}(attenuation factor)*

Remote command:
`CHANnel<m>:EATScale` on page 1346
`CHANnel<m>:EATTenuation` on page 1346

Impedance
Sets the impedance of the channel for power calculations and measurements.

Remote command:
`CHANnel<m>:IMPedance` on page 1340

Coupled to C1 / C2 / C3 / C4
Channel coupling sets the vertical settings of the coupled channels to the values of the active channel. If you want to have the same vertical settings for two or more channels, you can set them at once by coupling these channels.

Channel coupling affects all vertical settings that are adjusted in the "Settings" tab: vertical scale, position, offset, coupling, and ground.

Remote command:
`CHANnel<m>:CPLing` on page 1340

Filter characteristics
Selects the type of filter that is used. In the acquisition system, you can define the roll-off of the frequency response of the instrument.

Note: This setting does not apply to oscilloscopes with 4 GHz bandwidth.

"Brick wall" Ensures a maximally flat frequency response up to the instrument bandwidth. Also a maximally sharp rolloff above the instrument bandwidth to minimize the noise level.

"Gaussian" Ensures minimum overshoots in the recorded waveforms. Also a more flat rolloff above the instrument bandwidth with a slightly higher noise level than the Brick wall characteristic.

Remote command:
`CHANnel<m>:FILDesign` on page 1341

5.5.5 Digital filter setup

Access: "Menu" > "Vertical" > "Bandwidth" tab > "Digital Filter"

After processing by the A/D converter, the channel and trigger signals are digitized signals. These digitized signals can be filtered to reject high frequency - also known as Digital Signal Processing (DSP). You can filter the acquisition channels and the trigger channel signal.

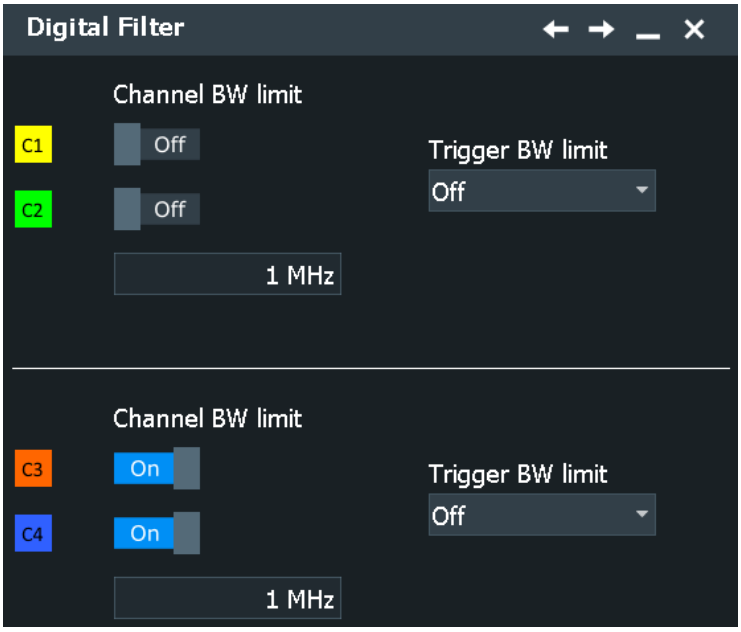
One filter is applied to a pair of channels. If you filter only on the input channels, you can apply different filters to the channels - one filter for channels 1 and 2 and another filter for channels 3 and 4.

If you filter the trigger channel, the same filter must be used for the input channels to ensure that all signals suit for analysis. The instrument offers only permitted combinations and triggers on the filtered signal.

In high definition mode, digital filter settings are enabled automatically. You can change the high definition bandwidth in the "Digital Filter" setup, which is applied to the channels.

Example:

"RF reject" for the trigger signal ensures that triggering is not caused by unexpected glitches.



Channel BW limit (On/Off)
Enables the DSP filter for the correspondig input channel.

Remote command:
`CHANnel<m>:DIGFilter:STATe` on page 1383

Channel BW limit
If "Trigger BW limit" is "Off", it sets the limit frequency of the lowpass filter for input channels. One filter is applied to a pair of channels.

If "Trigger BW limit" is "RF reject", it sets the limit frequency that is applied to the trigger channel and to the input channels enabled for filtering.

Remote command:
`CHANnel<m>:DIGFilter:CUToff` on page 1383
`TRIGger<m>:RFReject` on page 1384

Trigger BW limit
Selects the filter for the trigger channel. Other channels must use the same filter, or proceed unfiltered.

"Off"	The trigger signal is not filtered, and the acquisition channels can be filtered independently.
"RF reject"	Frequencies higher the "Channel BW limit" are rejected, lower frequencies pass the filter.

Remote command:
`TRIGger<m>:COUPling` on page 1383

5.6 Probes

With R&S RTO6 digital oscilloscopes, you can use various probe types. Mostly these probes are passive and active voltage probes. The "Probes" dialog provides all probe-relevant information.

The instrument can detect many probes and read out the probe-specific parameters, for example, bandwidth and attenuation.

In the "Probes" tab, you find all settings that are relevant for the connected probe.

Access: "Menu" > "Vertical" > "Probes"

The functionality on the "Probes" tab changes according to the type of the attached probe. Probes with Rohde & Schwarz probe interface (probe box), and also many other passive voltage probes, are recognized by the instrument. The R&S RTO6 reads out the main characteristics of the probe and displays them. Other probes cannot be detected, but their characteristics are known to the instrument. These known probes are called "Predefined probes". Probes that are not recognized automatically and not predefined are unknown probes, they require manual setting of measurement unit and attenuation.

Before you adjust the settings, select the correct channel tab.

Additional information is given in the "Probe Attributes" and "Calibration Results" tabs. For background information, see Chapter 5.1.4, "Probes", on page 115.

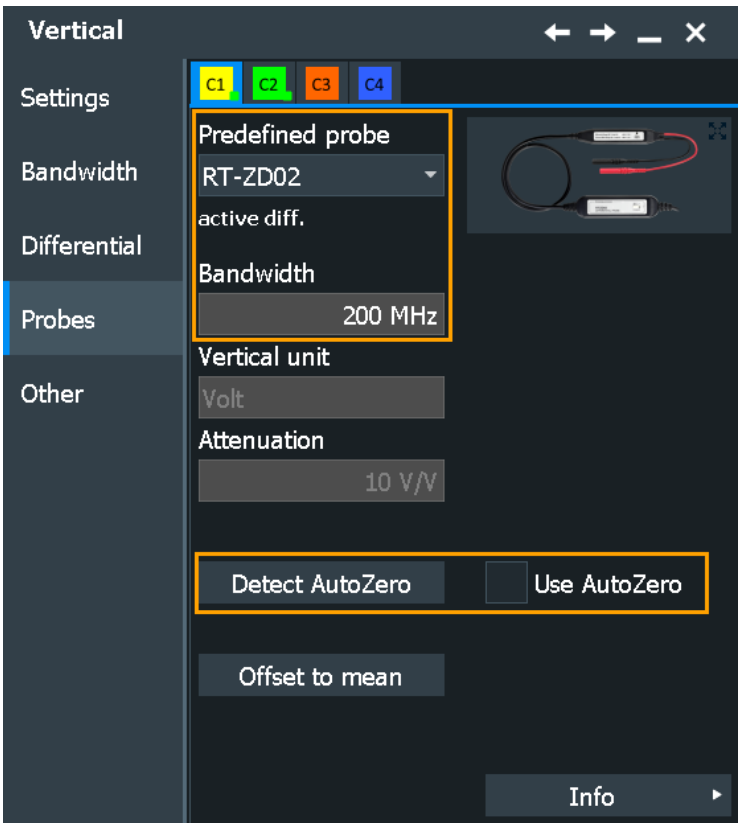
This chapter has the following sections:

- Shared probe settings....146
- Settings for the R&S probe interface (voltage probes)....147
- Setup for passive probes....150
- Setup for active voltage probes....152
- Modular probes....157
- Setup for predefined probes....162
- Setup for current probes....164
- Setup for unknown probes....167

- Probe attributes....168
- Calibration results....169
- Probe adapter R&S RT-Z2T....169

5.6.1 Shared probe settings

Some of the settings in the "Probes" tab are available for all probes.



An external attenuation can be set on the "Other" tab, see "External attenuation: Scale, Attenuation" on page 142.

The shared probe settings are:

Type, Name, Bandwidth....146
Detect AutoZero, Use AutoZero....147

Type, Name, Bandwidth
The fields show the characteristics of a recognized or predefined probe for information. If the instrument cannot recognize the probe, and the probe is not known, the "Type" is "None", and the other fields are empty.

Remote command:
`PROBe<m>:SETup:TYPE?` on page 1344
`PROBe<m>:SETup:NAME?` on page 1344
`PROBe<m>:SETup:BANDwidth?` on page 1345

Detect AutoZero, Use AutoZero

Differences in DUT and oscilloscope ground levels can cause larger zero errors, which affect the waveform. If the DUT is ground-referenced, the AutoZero function corrects the zero error of the probe to optimize measurement results at small signal levels. The validation limit depends on the probe attenuation because probes with high attenuation often have to compensate high offsets. AutoZero detects offset values even when the signal is out of the current measurement range.

To correct the zero error of voltage probes, short the signal pin and the ground pin together and connect them to the ground of the DUT. Then tap "Detect AutoZero". While the alignment is running, the instrument switches to DC coupling to display the waveform correctly.

To include the measured offset in measurement results, enable "Use AutoZero".

If a current probe is connected, the function demagnetizes the probe's sensor head and sets the waveform to zero position. See "Detect AutoZero" on page 167.

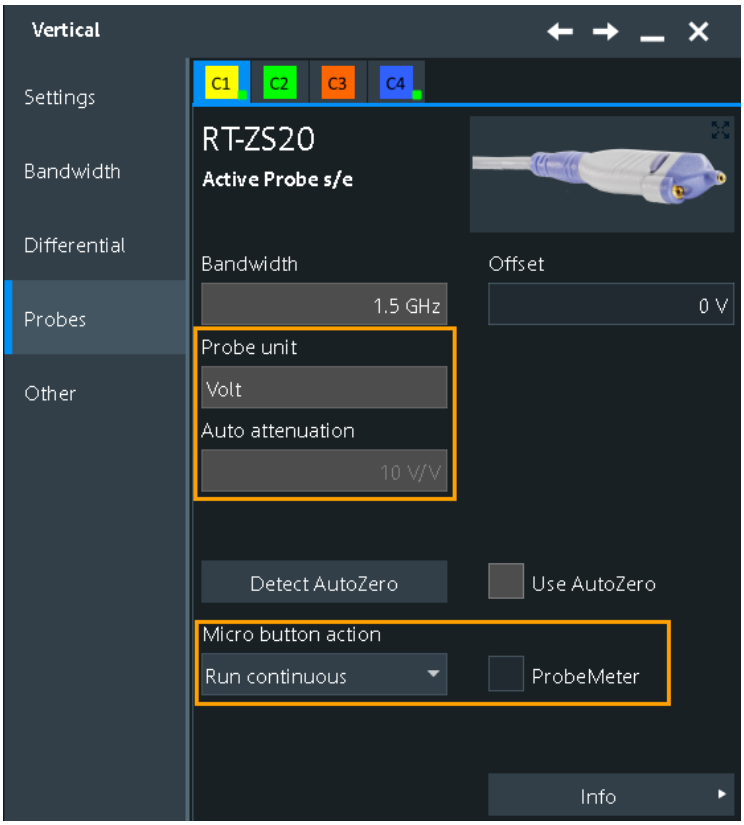
Remote command:

`PROBe<m>:SETup:OFFSet:AZERo` on page 1345

`PROBe<m>:SETup:OFFSet:USEautozero` on page 1346

5.6.2 Settings for the R&S probe interface (voltage probes)

Active voltage probes with Rohde & Schwarz probe interface provide special features: the micro button and the ProbeMeter. Furthermore, the R&S RTO6 can read out the attenuation of the probe.



The settings for active voltage probes with Rohde & Schwarz probe interface are:

Probe unit, Auto attenuation.....148
Micro button action.....148
ProbeMeter..... 149

Probe unit, Auto attenuation

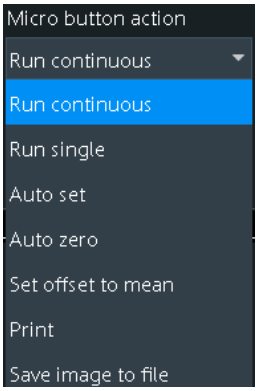
If the probe is recognized by the R&S RTO6, the instrument reads the attenuation unit and value from the probe an displays them.

Remote command:

`PROBe<m>:SETup:ATTenuation[:AUTO]?` on page 1345

Micro button action

Active voltage probes with Rohde & Schwarz probe interface have a configurable micro button on the probe head. Pressing this button, you start an action on the instrument directly from the probe. The button is disabled during internal automatic processes, for example, during self-alignment, autoset, and find level.



Select the action that you want to start from the probe:

"Run Continuous"
: Is the default assignment. Starts or stops the acquisition (same as Run Stop key).

"Run single"
: Starts a defined number of acquisitions (same as [Single] key).

"Auto set"
: Starts the autoset procedure (same as Autoset key).

"AutoZero"
: Starts an auto zero measurement, see "Detect AutoZero, Use AutoZero" on page 147.

"Set offset to mean"
: Performs an automatic compensation for a DC component of the input signal using the result of a background mean measurement. See: "Offset to mean" on page 164.

"Save image to file"
: Saves the current display as image according to the image settings. See also Chapter 12.3, "Screenshots", on page 475.

"No action"
: Select this option to prevent unwanted actions due to unintended usage of the micro button.

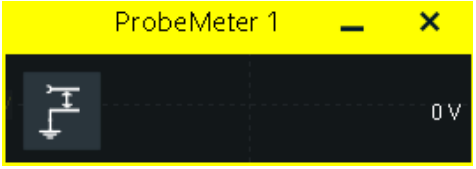
"Find trigger level"	Sets the trigger level automatically to *0.5 * (MaxPeak – MinPeak)*.
"Create report"	Creates and saves a report using the report settings. See also Chapter 12.4.1, "Report settings", on page 480.
"Probe mode"	Only available if a R&S RT-ZM modular probe is connected. Sets the measurement mode of the modular probe. See also "Probe mode" on page 158.
"Probe Setup"	Opens the "Probes Setup" dialog box.

Remote command:
`PROBe<m>:SETup:MODE` on page 1347

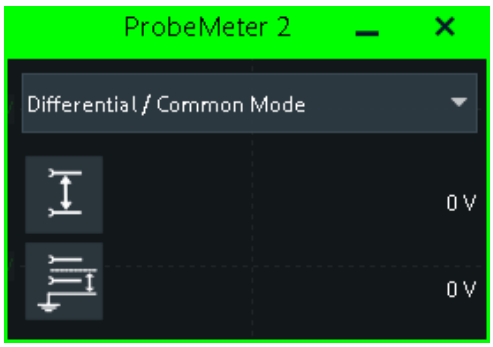
ProbeMeter

The integrated R&S ProbeMeter of active voltage probes with Rohde & Schwarz probe interface is a voltmeter. It measures DC voltages between the probe tip and ground connection or between the probe tips with very high precision. The R&S ProbeMeter enables ground-referenced measurements of voltages. The measurement is performed continuously and in parallel to the measurements of the oscilloscope.

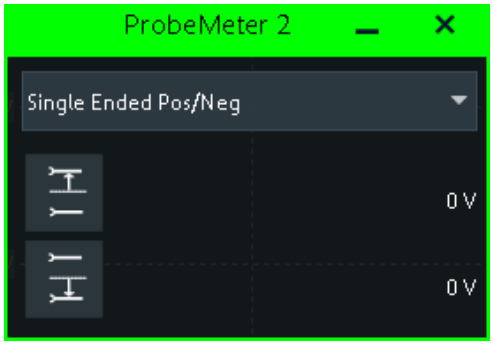
- **"Probemeter"**
  Select "Probemeter" to activate the integrated R&S ProbeMeter of active R&S probes. The measured voltages are displayed in the "ProbeMeter" result box on the screen.
- **ProbeMeter measurement results of single-ended active R&S probes**
  Measures the voltage between the probe tip and the ground.



- **ProbeMeter measurement results of differential and modular R&S probes**
  You can select the voltage to be measured by the differential active probe:
  – "Differential / Common Mode":
    Differential voltage is the voltage between the positive and negative signal sockets.
    Common mode voltage is the mean voltage between the signal sockets and the ground socket. It measures the voltage level relative to ground, for example, to check the operating voltage window.



  – "Single Ended Pos/Neg": Measures the voltage between the positive/negative signal socket and the ground.



The ProbeMeter always measures the common mode and differential voltages. Single-ended voltages are calculated values:
$V_p = V_{cm} + 0.5 * V_{in}$ and $V_n = V_{cm} - 0.5 * V_{in}$

Remote command:
`PROBe<m>:PMETer:VISibility` on page 1348
`PROBe<m>:SETup:DISPlaydiff` on page 1348
`PROBe<m>:PMETer:RESults:SINGle?` on page 1349
`PROBe<m>:PMETer:RESults:POSitive?` on page 1350
`PROBe<m>:PMETer:RESults:NEGative?` on page 1350
`PROBe<m>:PMETer:RESults:DIFFerential?` on page 1349
`PROBe<m>:PMETer:RESults:COMMon?` on page 1349

5.6.3 Setup for passive probes

Passive probes are the most widely used probes for oscilloscope measurements. Passive probes require compensation.

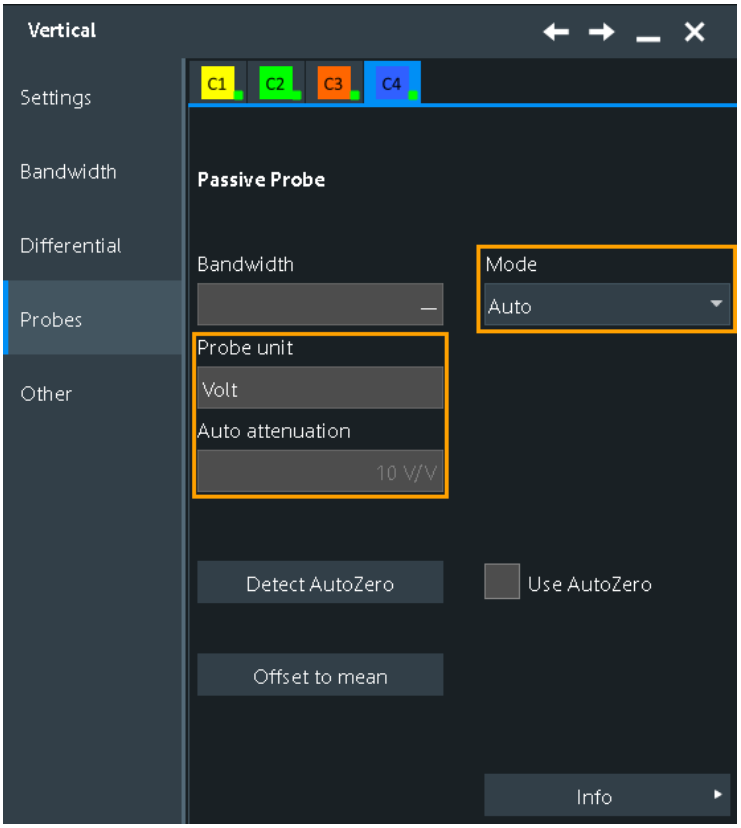


Figure 5-7: Probe setup for passive probe

The following shared probe settings are available:

- "Type, Name, Bandwidth" on page 146
- "Detect AutoZero, Use AutoZero" on page 147

If a passive probe is connected, the probe attenuation is read out and shown in the "Setup" tab:

- "Probe unit, Auto attenuation" on page 148

If you need to change the unit or attenuation, change the "Mode" to "Manual" and enter the correct values.

Mode..... 151
Vertical unit, Attenuation, Gain..... 152

Mode

Defines how the attenuation of a passive probe is set.

"Auto"	The instrument uses the values that are read out from the probe.
"Manual"	You can define the attenuation unit and value. See: "Vertical unit, Attenuation, Gain" on page 152

Remote command:

`PROBe<m>:SETup:ATTenuation:MODE` on page 1351

Vertical unit, Attenuation, Gain
If a predefined probe is connected and selected, the attenuation or gain values are shown.

For unknown probes and passive probes in manual mode, you can set user-defined values for unit, gain and attenuation.

If AC coupling is set, the attenuation of passive probes has no effect, and voltage is applied to the instrument with factor 1:1. Observe the voltage limits, otherwise you can damage the instrument.

Remote command:
`PROBe<m>:SETup:ATTenuation:UNIT` on page 1351
`PROBe<m>:SETup:ATTenuation:MANual` on page 1351
`PROBe<m>:SETup:GAIN:MANual` on page 1351

5.6.4 Setup for active voltage probes

Active voltage probes with Rohde & Schwarz probe interface have an integrated data memory that contains identification data and individual probe correction parameters. The R&S RTO6 can detect these probes and read out the data. Furthermore, these probes have a micro button and a ProbeMeter.

Active voltage probes that are offered by Rohde & Schwarz but not equipped with a Rohde & Schwarz probe interface are known to the R&S RTO6 as predefined probes, see Chapter 5.6.6, "Setup for predefined probes", on page 162.

The following shared probe settings are available:

- "Type, Name, Bandwidth" on page 146
- "Detect AutoZero, Use AutoZero" on page 147

Special features of the Rohde & Schwarz probe interface are described in these sections:

- "Probe unit, Auto attenuation" on page 148
- "Micro button action" on page 148
- "ProbeMeter" on page 149

The specific settings of Rohde & Schwarz active probes are described in the following chapters:

- R&S RT-ZS single-ended probes.....153
- R&S RT-ZD differential probes.....154
- R&S RT-ZPR power rail probes.....155
- R&S RT-ZHD high-voltage differential probes.....156

5.6.4.1 R&S RT-ZS single-ended probes

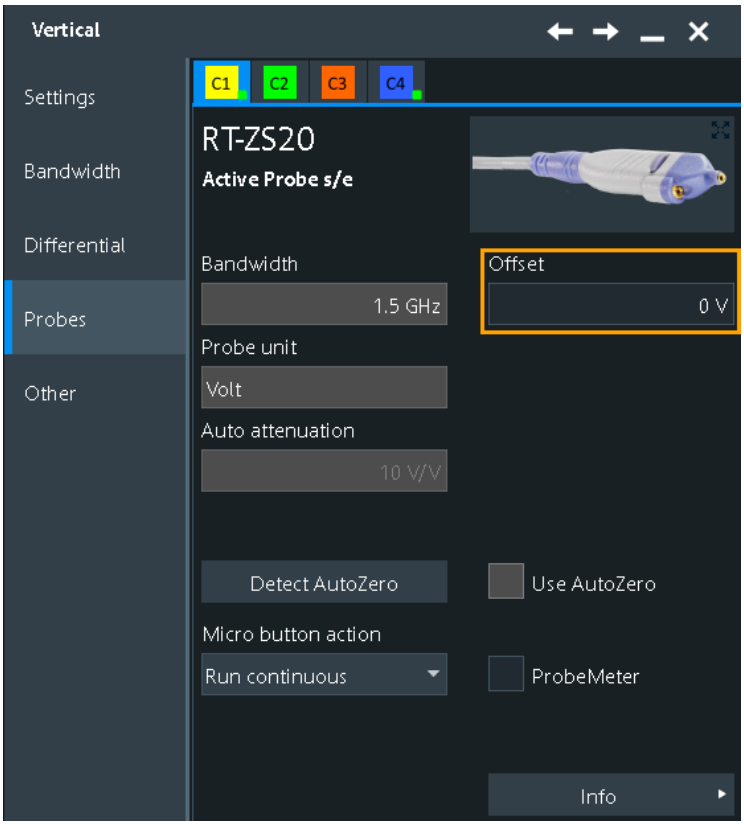


Figure 5-8: Probe setup for active single-ended probe R&S RT-ZS20

The only setting for R&S RT-ZS probes is the channel offset. See "Offset" on page 139.

5.6.4.2 R&S RT-ZD differential probes

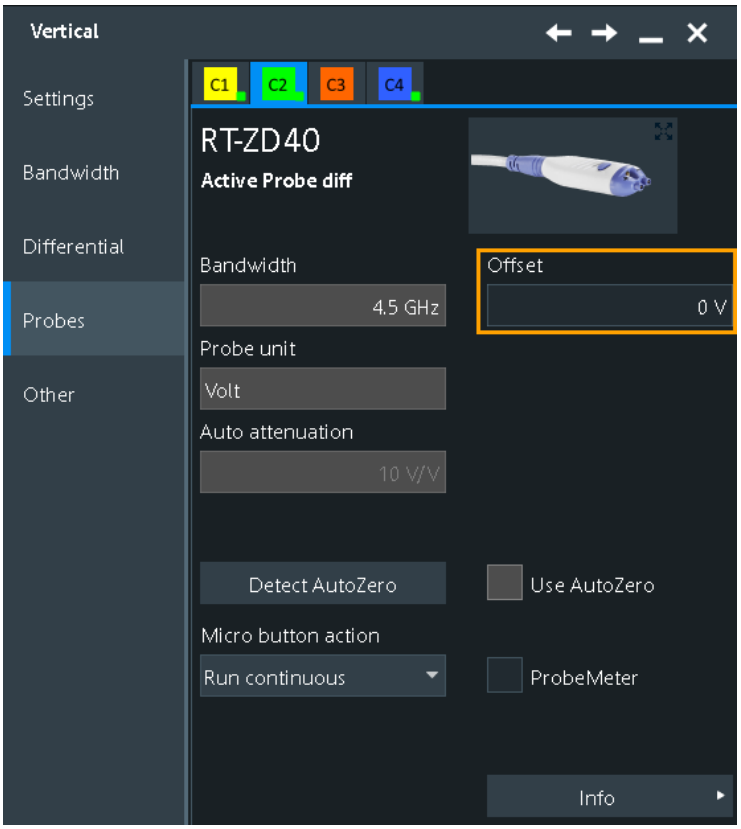


Figure 5-9: Probe setup for active differential probe R&S RT-ZD40

The offset is the differential offset. See "Offset" on page 139.

Specific settings for R&S RT-ZD probes are the following:

CM offset
Sets the common-mode offset to compensate for a common DC voltage applied to both input sockets (referenced to the ground socket). Offset compensation is particularly helpful for measurements on differential signals with high common mode levels, for example, current measurements using a shunt resistor. You can measure the common mode input voltage using the R&S ProbeMeter.

The setting is available for Rohde & Schwarz differential probes, and for modular probes in DM or CM mode (see "DM Offset, CM Offset, P Offset, N Offset" on page 159).

Remote command:
`PROBe<m>:SETup:CMOFfset` on page 1352

Probe attenuator RT-ZA15
If you use the external attenuator R&S RT-ZA15 together with one of the differential active probes R&S RT-ZD10/20/30, enable RT-ZA15 to include the external attenuation in the measurements.

Remote command:
`PROBe<m>:SETup:ZAXV` on page 1352

5.6.4.3 R&S RT-ZPR power rail probes

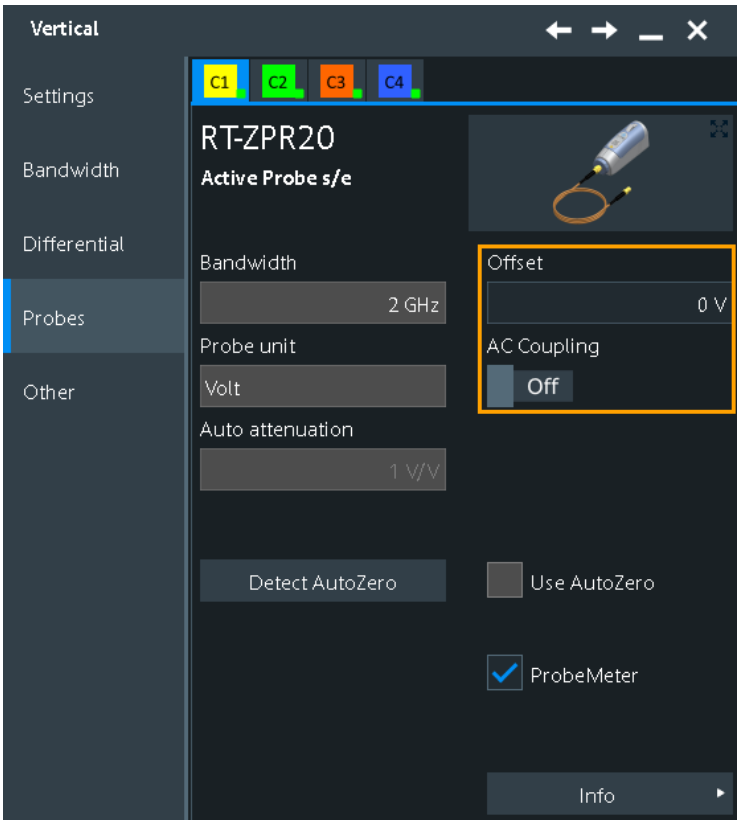


Figure 5-10: Probe setup for power rail probe R&S RT-ZPR

The offset is the channel offset. See "Offset" on page 139.

If the ProbeMeter is active, an additional function is provided in the "ProbeMeter" result box:

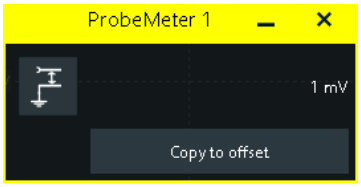


Figure 5-11: ProbeMeter result box for

Specific settings for R&S RT-ZPR probes are the following:

AC Coupling
Enables AC coupling in the R&S RT-ZPR power rail probes, which removes DC and very low-frequency components. The R&S RT-ZPR probe requires 50 Ω input termination, for which the channel AC coupling is not available. The probe setting allows AC coupling also at 50 Ω inputs.

Remote command:
`PROBe<m>:SETup:ACCoupling` on page 1353

Copy to offset
Sets the measured ProbeMeter value as offset. Thus, the value is considered in measurements.

Remote command:
`PROBe<m>:SETup:ADVanced:PMToffset` on page 1353

5.6.4.4 R&S RT-ZHD high-voltage differential probes

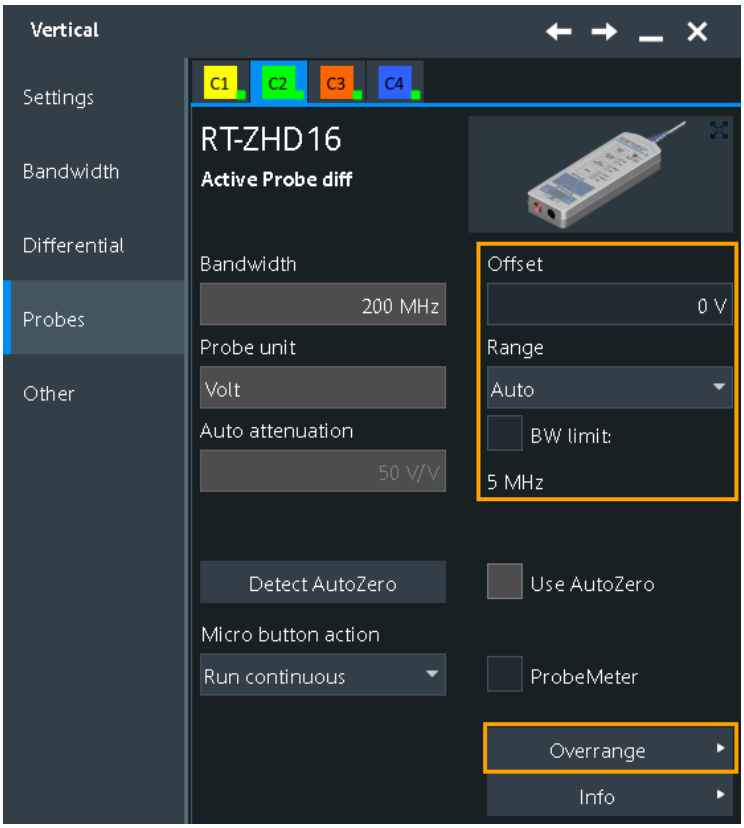


Figure 5-12: Probe setup for R&S RT-ZHD probes

The offset is the differential offset. See "Offset" on page 139.

Specific settings for R&S RT-ZHD probes are the following:

Range
Sets the voltage range of a R&S RT-ZHD probe. You can set the range on the probe control box or at the oscilloscope.

"Auto"	The voltage range is set only at the oscilloscope with "Vertical scale".
"Low"	Sets the lower voltage range of the connected probe. The selected value is shown in "Auto Attenuation".
"High"	Sets the higher voltage range of the connected probe. The selected value is shown in "Auto Attenuation".

Remote command:
`PROBe<m>:SETup:ADVanced:RANGe` on page 1353

BW limit
Activates the lowpass filter in the probe control box and displays the used limit. You can also set the filter directly on the probe control box.

Remote command:
`PROBe<m>:SETup:ADVanced:FILTer` on page 1354

Buzzer
Activates the acoustic overrange warning in the probe control box. You can also activate the sound directly on the probe control box.

Remote command:
`PROBe<m>:SETup:ADVanced:AUDioverload` on page 1354

Negative, Positive, Differential Overrange
The color turns red if the voltage exceeds the probe range. The indicators are also available on the probe control box.

5.6.5 Modular probes

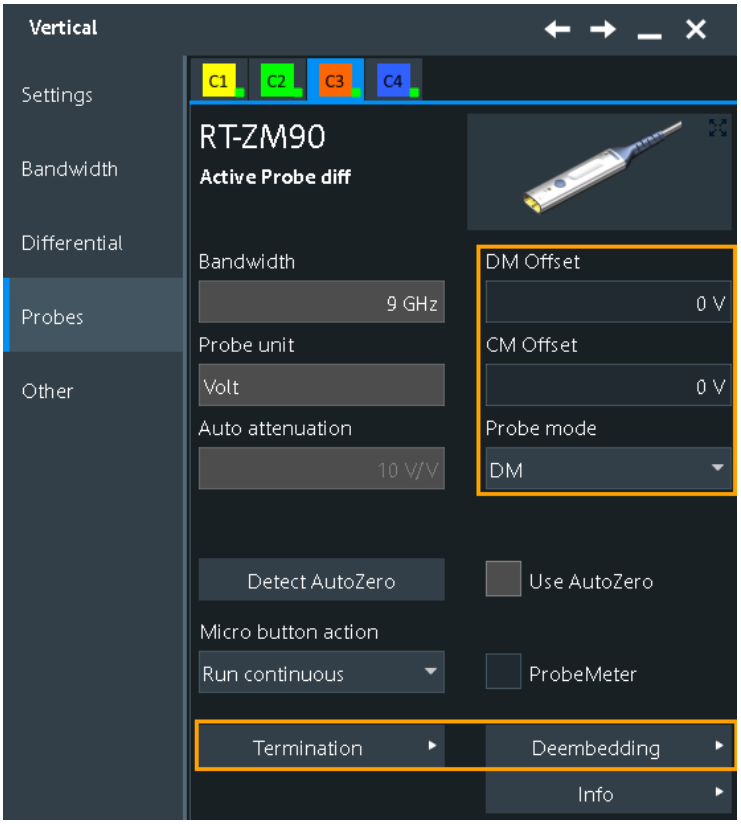
The probes of the R&S RT-ZM family are modular probes. They have a probe head and a probe amplifier connected by a cable, and various probe tip modules and tip cables for different applications. R&S RT-ZM probes are equipped with Rohde & Schwarz probe interface, and provide special features: ProbeMeter, micro button, and a wide offset compensation range.

The following shared probe settings are available:

- "Type, Name, Bandwidth" on page 146
- "Detect AutoZero, Use AutoZero" on page 147

Special features of the Rohde & Schwarz probe interface are described in these sections:

- "Probe unit, Auto attenuation" on page 148
- "Micro button action" on page 148
- "ProbeMeter" on page 149



- Setup parameters of modular probes....158
- Deembedding for modular probes....159
- Termination voltage with R&S RT-ZMA40 SMA module....161

5.6.5.1 Setup parameters of modular probes

The basic setup parameters of all modular probes are the measurement mode and the offset settings.

Access: "Menu" > "Vertical" > "Probes"

Make sure to adjust also the deembedding settings under "Probe Deembedding" (see Chapter 5.6.5.2, "Deembedding for modular probes", on page 159).

Probe mode
Sets the measurement mode of modular probes.

The modular probes of the R&S RT-ZM family have a multi-mode function. You can switch between single-ended, differential and common mode measurements without reconnecting or resoldering the probe. You can set the probe mode in the dialog box, and you can assign the probe mode setting to the micro button.

If you use the R&S RT-ZMA30 browser module, only DM measurements are possible because this module has no ground connector.

The measurement modes are:

"DM" Differential mode input voltage (V_{dm}), the voltage between the positive and negative input terminal.

$$V_{dm} = V_p - V_n$$

"CM" Common mode input voltage (V_{cm}), the mean voltage between the positive and negative input terminal vs. ground.

$$V_{cm} = \frac{V_p + V_n}{2}$$

"P" Positive single-ended input voltage (V_p). The voltage between the positive input terminal and ground.

"N" Negative single-ended input voltage (V_N). The voltage between the negative input terminal and ground.

Remote command:
`PROBe<m>:SETup:PRMode` on page 1354

DM Offset, CM Offset, P Offset, N Offset
Compensate offset voltages. Available offsets depend on the selected probe mode.

The offset of the selected probe mode is used as channel offset and considered automatically for correction. For example, in CM mode, the common mode offset is used as channel offset. See also: "Offset" on page 139.

"DM Offset" Compensates a DC voltage applied between the positive (V_p) and the negative (V_n) input terminal at the probe tip.

"CM Offset" Compensates a DC voltage applied to both input terminals referenced to ground. See also: "CM offset" on page 154.

"P Offset" Compensates a DC voltage applied to the positive input terminal (V_p) referenced to ground.

"N Offset" Compensates a DC voltage applied to the negative input terminal (V_n) referenced to ground.

Remote command:
`PROBe<m>:SETup:DMOFfset` on page 1355
`PROBe<m>:SETup:CMOFfset` on page 1352
`PROBe<m>:SETup:NOFFset` on page 1356
`PROBe<m>:SETup:POFFset` on page 1356

5.6.5.2 Deembedding for modular probes

Deembedding removes the parasitic effects of the measurement setup from the measured signal. These effects are typically increasing when signal frequency increases. Thus, deembedding is useful or even necessary when measuring signals of 4 GHz frequency or higher.

Access: "Vertical" menu > "Probe Setup" > "Probe Deembedding"

Select probe tip module, Pin spacing
Selects the tip module that is used for measurement.

If R&S RT-ZMA40 is selected, choose also the used submodule: semi-rigid cables or none.

For the browser module R&S RT-ZMA30, measure the space between the pins, and select the appropriate value.

Remote command:
`PROBe<m>:DEEMbedding:TIPModule[:SELect]` on page 1358
`PROBe<m>:DEEMbedding:TIPModule:ZMA<n>:SUBModule` on page 1359

Probing, Terminating
Sets the probing mode.

Select "Probing" if you use a high-ohmic probe and measure on an existing line in parallel to the load.

Select "Terminating" if the measuring equipment is the load of the line.

Remote command:
`PROBe<m>:DEEMbedding:MODE` on page 1360
`DEEMbedding<m>:COMPonent<n>:MODE` on page 2564

Effective bandwidth
Sets the maximum bandwidth for probe deembedding until which the signal is corrected. This maximum value is the minimum bandwidth value of the probe bandwidth, tip module bandwidth, and oscilloscope bandwidth. It cannot be higher than the highest frequency in a used S-parameter file.

Consider that most tip modules support the full bandwidth of the probe, but some tip modules have limited bandwidth.

If deembedding option R&S RTO6-K121 is active, the instrument uses the effective bandwidth that is set in the "Deembedding" > "Setup" dialog box.

Remote command:
`PROBe<m>:DEEMbedding:BANDwidth` on page 1359

Probe mode
See "Probe mode" on page 158.

Remove probe load
If "Probing" is selected, you can remove the loading of the probe.

If the probe loading is removed, you see the signal that would be at the measurement point if the probe's input impedance is ideal (infinite impedance).

Without removing the loading, you see the real signal at the measurement point, including the probe loading.

Remote command:
`PROBe<m>:DEEMbedding:REMProbeload` on page 1360
`DEEMbedding<m>:COMPonent<n>:LOAD:REMProbeload` on page 2565

Source impedance
Shows the source impedance of the DUT. If an R&S RT-ZM probe is connected, the value depends on the selected probe mode: common mode, differential, or single-ended measurement.

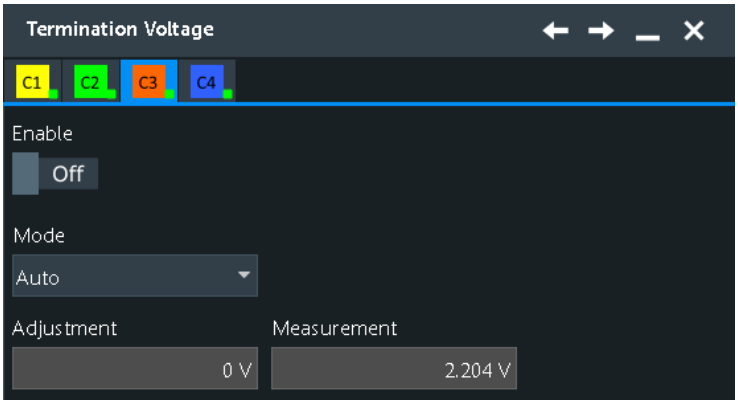
Remote command:
`DEEMbedding<m>:COMPonent<n>:LOAD:IMPedance?` on page 2565
`PROBe<m>:DEEMbedding:LOAD:IMPedance` on page 1361

5.6.5.3 Termination voltage with R&S RT-ZMA40 SMA module

Termination voltage is relevant if you use the R&S RT-ZMA40 SMA module. The SMA module applies a termination voltage (±4 V) to the DUT to enable measurements against a common mode DC voltage instead of ground. This measurement is required for many digital signal standards.

The termination voltage can be controlled by the oscilloscope. Therefore, connect the V_T terminal of the R&S RT-ZM probe amplifier to the V_T terminal of the R&S RT-ZMA40 SMA module using the red DC lead (see R&S RT-ZM User Manual). The required termination voltage is measured and adjusted automatically, but can also be set manually.

Access: "Menu" > "Vertical" > "Probes" > "Termination"



Enable
Activates control of the termination voltage.

Remote command:
`PROBe<m>:SETup:TERM:STATe` on page 1356

Mode
In "Auto" mode, the instrument uses the measured common mode voltage for termination.

In "Manual" mode, you can enter the voltage to be used for termination. Use the manual mode if you know the common mode voltage of the DUT.

Remote command:
`PROBe<m>:SETup:TERM:MODE` on page 1357

Adjustment
Sets the voltage to be used for termination to DC voltage.

Remote command:
`PROBe<m>:SETup:TERM:ADJust` on page 1357

Measurement
Shows the measured common mode voltage.

Remote command:
`PROBe<m>:SETup:TERM:MEASure?` on page 1357

5.6.6 Setup for predefined probes

Probes that cannot be detected, but their characteristics are known to the R&S RTO6 are called "Predefined probes".

The following shared probe settings are available:

- "Type, Name, Bandwidth" on page 146
- "Detect AutoZero, Use AutoZero" on page 147

The probe attenuation of the selected probe is also shown in the "Probes" tab:

- "Probe unit, Auto attenuation" on page 148

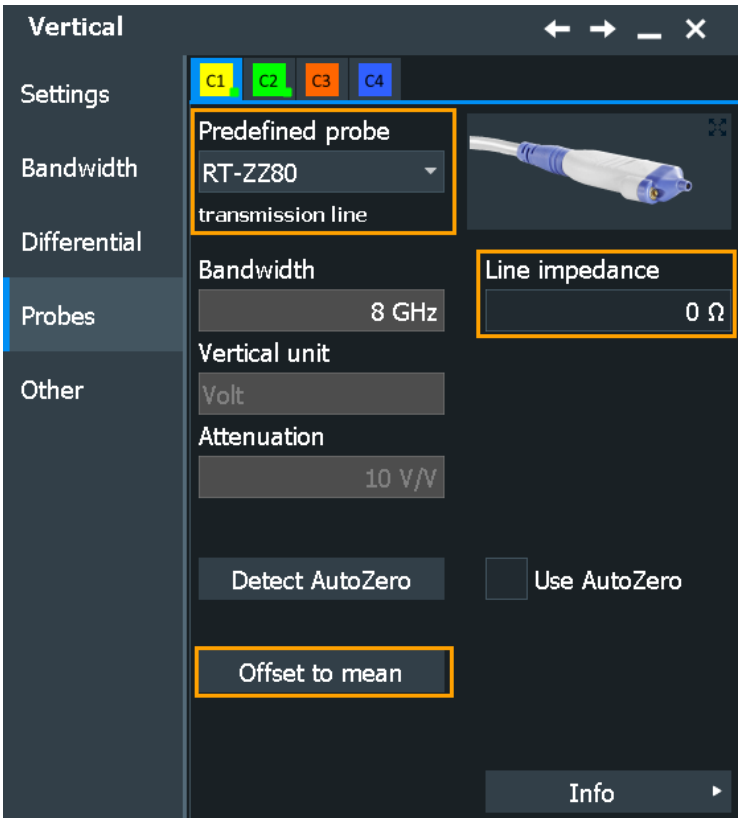


Figure 5-13: Probe setup for transmission line probe R&S RT-ZZ80

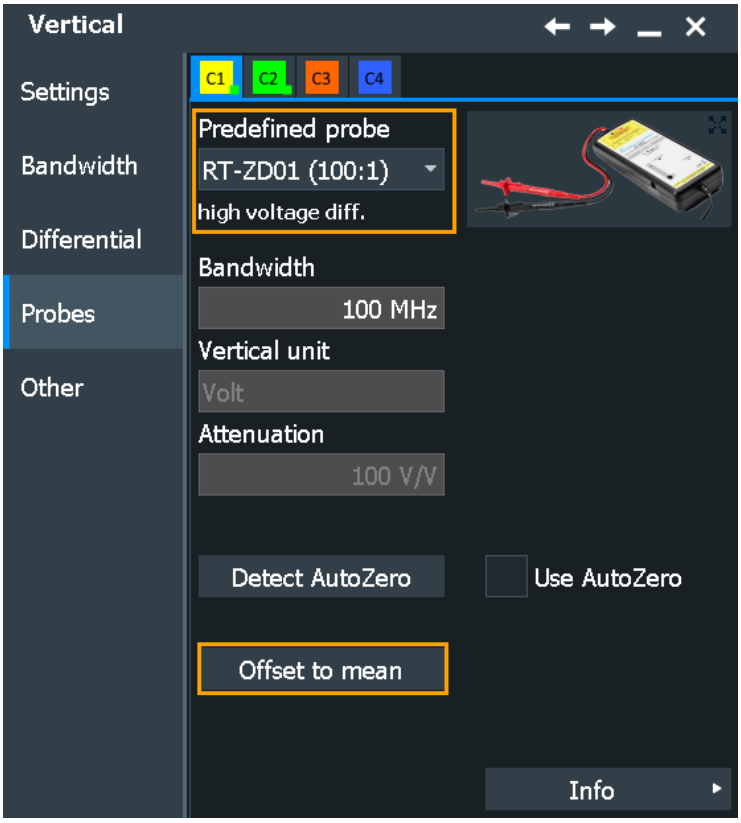


Figure 5-14: Probe setup for R&S RT-ZD01

Specific settings for predefined probes are the following:

Predefined probe.....163
Offset to mean.....164
Line impedance.....164

Predefined probe

List of probes that are known to the instrument. The instrument lists only probes that match the instrument coupling and, if connected, also the used adapter.

Select the used probe on the list. The corresponding "Vertical unit" and the "Attenuation" or "Gain" are shown.

All other unrecognized probes that are not listed, are unknown probes. For these probes, set "Predefined probe" to "None". See Chapter 5.6.8, "Setup for unknown probes", on page 167.

Remote command:

`PROBe<m>:SETup:ATTenuation:DEFProbe` on page 1361

Offset to mean
Performs an automatic compensation for a DC component of the input signal using the result of a background mean measurement. The result is shown in "Offset". The function is probe-independent and supports quick and convenient measurements of input signals with different DC offsets. It detects offset values even when the signal is out of the current measurement range. It also sets the zero level to the determined DC offset in the middle of the screen and thus prevents clipping of the waveform.

Remote command:
`PROBe<m>:SETup:OFFSet:TOMean` on page 1362

Line impedance
If the transmission line probe R&S RT-ZZ80 is selected, enter the impedance of the measured line.

The actual attenuation of the transmission line probe depends on the impedance of the line Z_0:

$Attenuation = 10 + Z_0 / 100$

The instrument uses the actual attenuation to determine the measurement values.

5.6.7 Setup for current probes

The setup and adjustment of current probes depends on the output connector of the probe: BNC or Rohde & Schwarz probe box.

The following shared probe settings are available:

- "Type, Name, Bandwidth" on page 146
- "Detect AutoZero, Use AutoZero" on page 147

Current probes R&S RT-ZCxx

The current probes **R&S RT-ZCxx** have BNC connectors. They are known to the R&S RTO6 as predefined probes, see Chapter 5.6.6, "Setup for predefined probes", on page 162. Demagnetizing and zero adjustment is done on the probe, see the probe's User Manual for details. Make sure to demagnetize and adjust the probe before taking measurements.

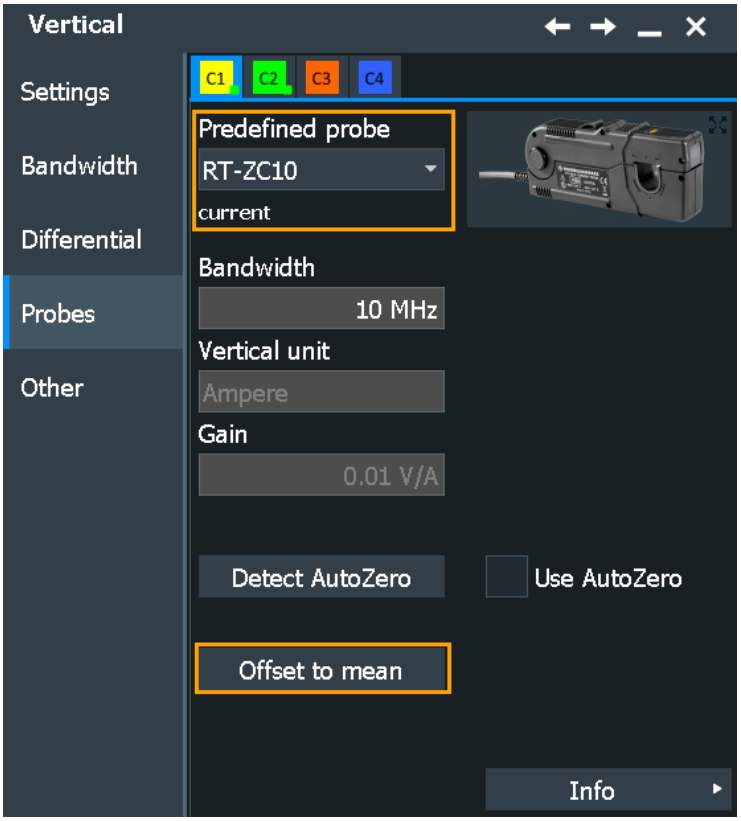


Figure 5-15: Probe setup for current probes R&S RT-ZC10

Current probes R&S RT-ZCxxB

Current probes **R&S RT-ZCxxB** have a Rohde & Schwarz probe interface; they are powered and remotely controlled by the oscilloscope.

When the probe is connected, demagnetization is performed automatically.

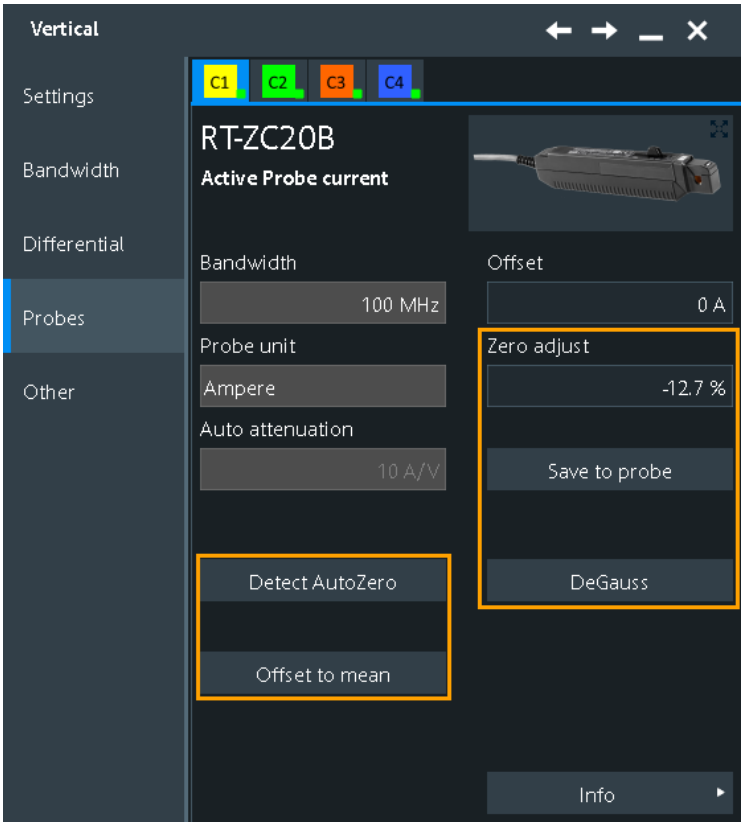


Figure 5-16: Probe setup for current probes R&S RT-ZC20B

For all current probes, attenuation or gain is shown in the "Setup" tab, and you can set the offset to mean:

- "Probe unit, Auto attenuation" on page 148
- "Offset to mean" on page 164

Current probes R&S RT-ZCxxB are adjusted by the following functions:

DeGauss..... 166
Detect AutoZero..... 167
Zero adjust..... 167
Save to probe..... 167

DeGauss

Demagnetizes the core if it has been magnetized by switching the power on and off, or by an excessive input. Always carry out demagnetizing before measurement.

The demagnetizing process takes about one second. During demagnetizing, a demagnetizing waveform is displayed.

Demagnetizing is done automatically when R&S RT-ZCxxB is connected to the oscilloscope, or when "Detect AutoZero" is performed.

Remote command:
`PROBe<m>:SETup:DEGauss` on page 1362

Detect AutoZero
If a current probe is connected, the function demagnetizes the probe's sensor head and sets the waveform to zero position to correct the error offset. Thus, it compensates for the remanence and offset caused by temperature drift.

For R&S RT-ZCxxB probes, the determined "Zero adjust" value is displayed and can be saved in the probe head.

See also "Detect AutoZero, Use AutoZero" on page 147.

Remote command:
`PROBe<m>:SETup:OFFSet:AZERo` on page 1345

Zero adjust
Zero adjust corrects the effect of an offset caused by temperature drift, and compensates for the remanence. The setting is only available if DC coupling is set.

To set the waveform to zero level by the instrument, use "Detect AutoZero". The detected value is displayed.

Alternatively, you can adjust the value manually until the waveform is set to zero level. Make sure to demagnetize the probe before zero adjustment.

The value is given in percent of the maximum range, which is internally defined. The actual setup range depends on the temperature drift, the measured current and other variables, and it can change over time. If you measure high currents, the probe core magnetizes, which impairs the measurement results. Therefore, repeat "Detect AutoZero" before the measurement.

Remote command:
`PROBe<m>:SETup:OFFSet:ZADJust` on page 1363

Save to probe
Saves the "Zero adjust" value in the probe box. If you connect the probe to another channel or to another R&S RTx oscilloscope, the value is read out again, and you can use the probe without further adjustment.

Remote command:
`PROBe<m>:SETup:OFFSet:STPRobe` on page 1362

5.6.8 Setup for unknown probes

If the R&S RTO6 cannot detect the probe, and the probe is not a predefined one, you can set the probe parameters manually.

The following shared probe settings are available:

- "Type, Name, Bandwidth" on page 146
- "Detect AutoZero, Use AutoZero" on page 147

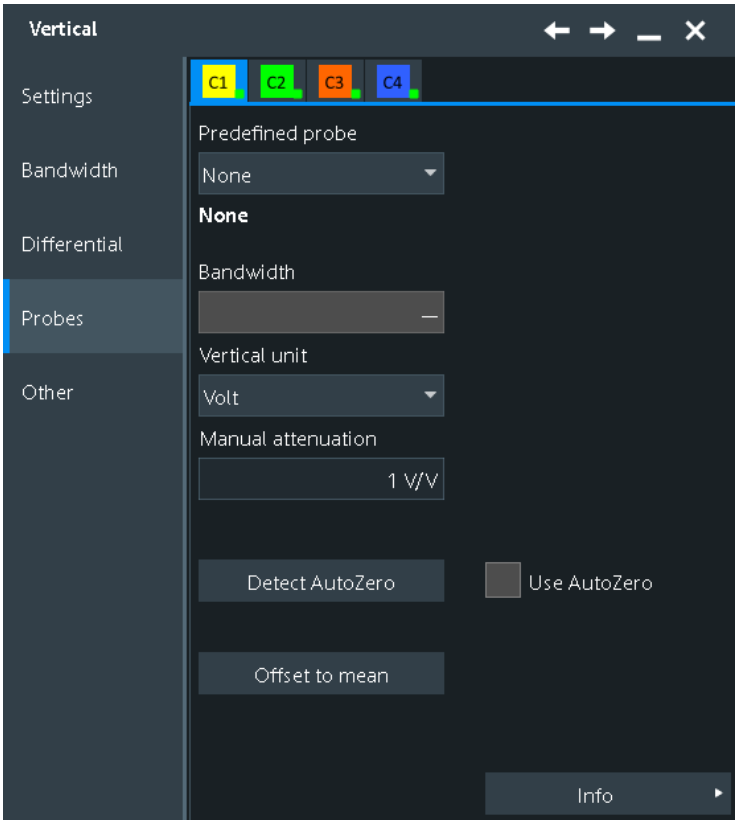


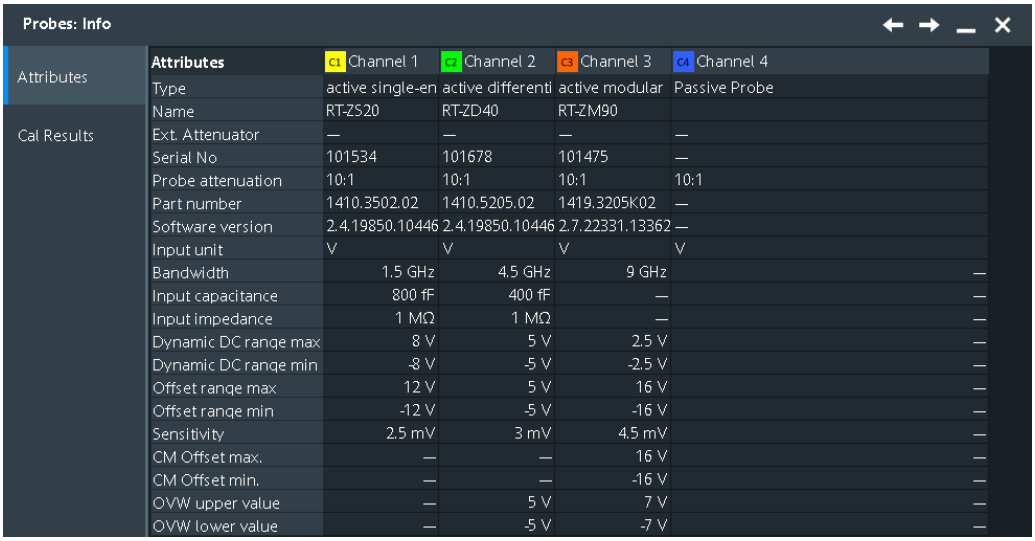
Figure 5-17: Probe setup for an unknown probe

Set the unit and the attenuation or gain of the probe: "Vertical unit, Attenuation, Gain" on page 152.

5.6.9 Probe attributes

To check the attributes of the connected probes, select "Info" in the "Vertical" > "Probes" tab.

For a specification of the probe parameters, refer to the data sheet.



Attributes	C1 Channel 1	C2 Channel 2	C3 Channel 3	C4 Channel 4
Type	active single-en	active differenti	active modular	Passive Probe
Name	RT-ZS20	RT-ZD40	RT-ZM90	
Ext. Attenuator	—	—	—	—
Serial No	101534	101678	101475	—
Probe attenuation	10:1	10:1	10:1	10:1
Part number	1410.3502.02	1410.5205.02	1419.3205K02	—
Software version	2.4.19850.10446	2.4.19850.10446	2.7.22331.13362	—
Input unit	V	V	V	V
Bandwidth	1.5 GHz	4.5 GHz	9 GHz	—
Input capacitance	800 fF	400 fF	—	—
Input impedance	1 MΩ	1 MΩ	—	—
Dynamic DC range max	8 V	5 V	2.5 V	—
Dynamic DC range min	-8 V	-5 V	-2.5 V	—
Offset range max	12 V	5 V	16 V	—
Offset range min	-12 V	-5 V	-16 V	—
Sensitivity	2.5 mV	3 mV	4.5 mV	—
CM Offset max.	—	—	16 V	—
CM Offset min.	—	—	-16 V	—
OVW upper value	—	5 V	7 V	—
OVW lower value	—	-5 V	-7 V	—

Remote commands:

- `PROBe<m>:ID:SWVersion?` on page 1363
- `PROBe<m>:ID:PRDate?` on page 1363
- `PROBe<m>:ID:PARTnumber?` on page 1364
- `PROBe<m>:ID:SRNumber?` on page 1364
- `PROBe<m>:SETup:CAPacitance?` on page 1364
- `PROBe<m>:SETup:IMPedance?` on page 1364

5.6.10 Calibration results

To check the attributes of the connected probes, select "Info" > "Cal Results" in the "Probes" tab.

Calibration	C1 Channel 1	C2 Channel 2	C3 Channel 3	C4 Channel 4
Probe group delay	5.409 ns	5.3583 ns	5.5472 ns	0 s
Attenuation	10.5803625:1	10.40935:1	10.7777:1	10:1

5.6.11 Probe adapter R&S RT-Z2T

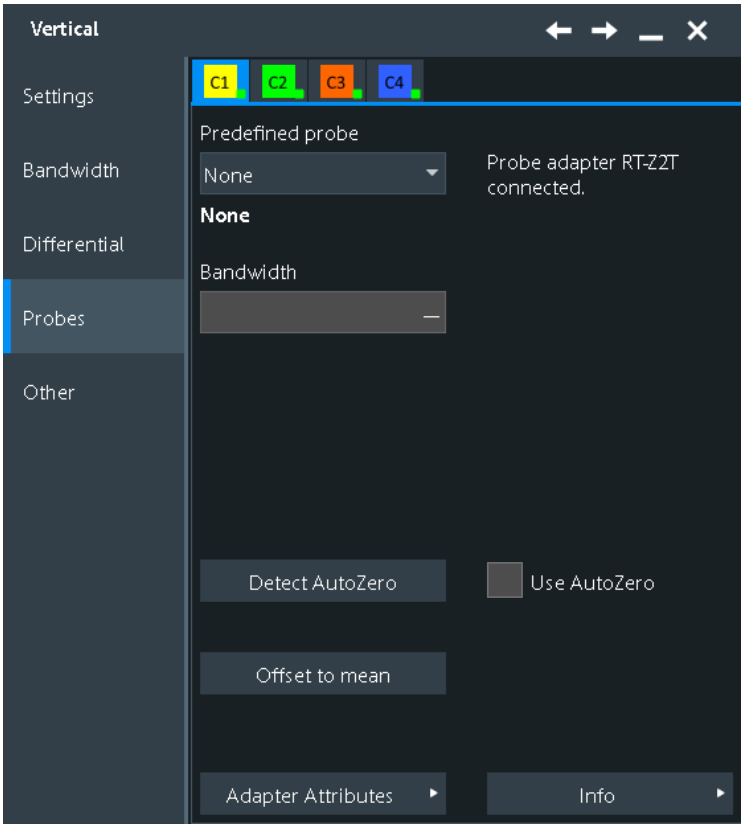
Using the R&S RT-Z2T probe interface adapter, you can connect selected Tektronix active probes with TekProbe BNC™ level II interface.

1. Connect the R&S RT-Z2T adapter to the channel input.
2. Connect the probe to the adapter.

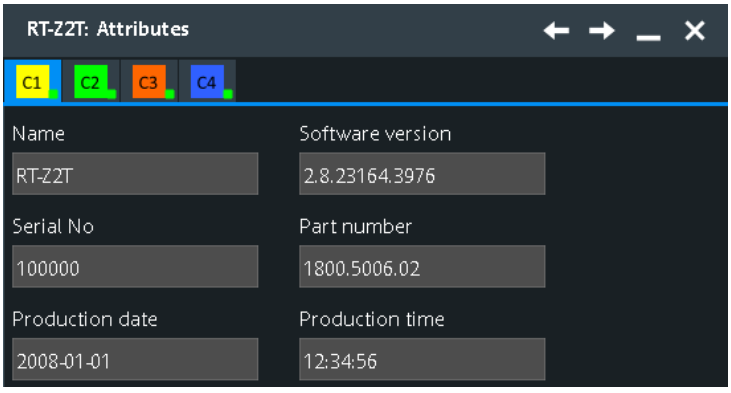
   The instrument identifies the adapter. The R&S RTO6 lists the supported Tektronix probes as "Predefined probe".

3. Select "Menu" > "Vertical" > "Probes".

   The dialog shows that the R&S RT-Z2T probe interface adapter is connected.



4. To see the information on the adapter, select "Adapter Attributes".



5. Select the "Predefined probe" in the "Probes" tab.
   See also: Chapter 5.6.6, "Setup for predefined probes", on page 162.

Remote commands:

- `PROBe<m>:SETup:ADAPter?` on page 1365
- `PROBe<m>:SETup:ATTenuation:TDEFprobe` on page 1365

5.7 R&S RT-ZVC probe

With the R&S RTO6 and option R&S RTO6-B1E, you can use the R&S RT-ZVC multi-channel power probe. It has an integrated 2- or 4-channel amperemeter and 2- or 4-channel voltmeter. The probe provides parallel measurements of analog or digital, voltage/current signals with excellent 18-bit resolution.

For more information on the R&S RT-ZVC probe, see also its user manual.

Source Channels

You can simultaneously connect a R&S RT-ZVC and a R&S RT-ZL04 to the R&S RTO6, but no parallel operation on screen is possible.

You can acquire and measure the R&S RT-ZVC or R&S RT-ZL04 together with the analog input channels. They are running on the same horizontal scale.

If an amperemeter or voltmeter channel is activated, it can be displayed on the screen and used as a source for:

- Cursor measurements
- Automatic measurements, and also histogram, limit checks, longterm, track, quick measurements
- Mask testing
- Basic and advanced maths
- Reference waveform
- Trigger type "Edge"
- Search: all search events
- XY diagrams
- Data export

All features of the base unit and the R&S RTO6-K37 option are supported.

Data export

You can save the data of the amperemeter and voltmeter channels to an XML, CSV, or BIN file. One channel per file can be saved. Files in BIN format can be reloaded to the R&S RTO6 as reference waveforms.

See also:

- Chapter 12.2.7, "Saving and loading waveform data", on page 473
- Chapter 12.2.1, "Waveform export files", on page 453

Remote commands for export to file:

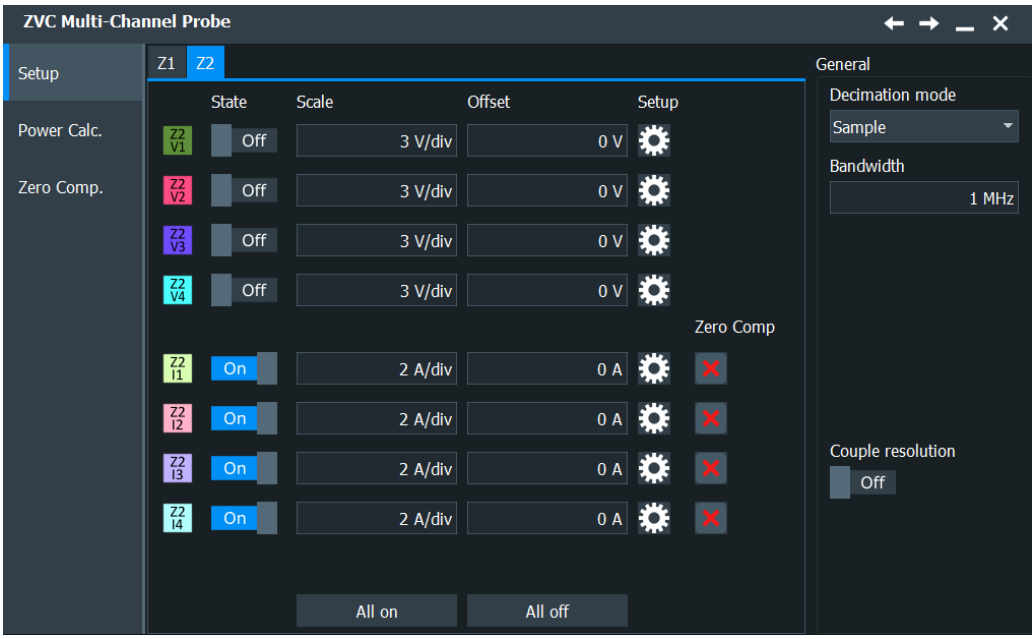
- `EXPort:WAVeform:SOURce` on page 1649
- `EXPort:WAVeform:NAME` on page 1651
- `EXPort:WAVeform:SAVE` on page 1651

5.7.1 R&S RT-ZVC overview

5.7.1.1 Setup

Access: [App Cockpit]> "Analysis" > "Multi-channel probe" > "Setup" tab

Make sure that the tab of the correct probe is selected on the left side.



State
Enables the corresponding channel of the probe. The number of available channels depend on the characteristics of your multi-channel power probe.

Remote command:
`ZVC:Z<m>:V<n>[:STATe]` on page 1375
`ZVC:Z<m>:I<n>[:STATe]` on page 1372

Scale
Defines the vertical scale for the channel in Volts per division. Increasing the scale compresses the display of the signal. Within a given operation range, modifying the scale is equivalent to scaling a display range.

For the voltmeter channels, the Scale, the Offset and the "Position" specify the operating range of the voltmeter.

Remote command:
`ZVC:Z<m>:V<n>:SCALe` on page 1374
`ZVC:Z<m>:I<n>:SCALe` on page 1370

Offset
The vertical center of the selected channel is shifted by the offset value and the signal is repositioned within the diagram area. Negative offset values move up the waveform, positive values move it down.

Within a given operation range, modifying the offset is equivalent to moving vertically the display range. The offset can only be modified such that the display range reaches at most the limits of the operation range.

For the voltmeter channels, the Scale, the Offset and the "Position" specify the operating range of the voltmeter.

Remote command:
`ZVC:Z<m>:V<n>:OFFSet` on page 1373
`ZVC:Z<m>:I<n>:OFFSet` on page 1368

Setup
Opens the voltage or current settings for the selected channel. See Chapter 5.7.2, "ZVC voltage settings", on page 176 and Chapter 5.7.3, "ZVC current settings", on page 177.

Zero Comp
Shows the status of the zero compensation of each current channel. The green checkmark indicates that the zero offset is compensated automatically. To adjust the settings, tap the icon. For details, see Chapter 5.7.1.3, "Zero compensation", on page 175.

Remote command:
`ZVC:Z<m>:I<n>:ZERComp:STATe?` on page 1377

All on
Enables all available channels.

All off
Disables all available channels.

Decimation mode
Selects the decimation mode for all R&S RT-ZVC probes. Decimation reduces the data stream of the ADC to a stream of waveform points with lower sample rate and a less precise time resolution.

"Sample"	One of n samples in a sample interval of the ADC is recorded as waveform point, the other samples are discarded. The time between the two adjacent waveform points is exactly the resolution. Very short glitches might remain undiscovered by this method.
"Peak detect"	The minimum and the maximum of n samples in a sample interval are recorded as waveform points, the other samples are discarded.
"High res"	The average of n sample points is recorded as one waveform sample. Averaging reduces the noise, the result is a more precise waveform with higher vertical resolution. The high measurement resolution is suitable for high accuracy measurements of instantaneous values.

Remote command:
`ZVC:TYPE` on page 1367

Bandwidth
Sets the bandwidth limit of all R&S RT-ZVC probes. The bandwidth specifies the maximum frequency at which a purely sinusoidal signal is still transferred at 89 % (1 dB) of its amplitude.

The bandwidth of some current channels is restricted to 300 KHz due to their vertical settings.

Remote command:
`ZVC:BANDwidth` on page 1367

Couple resolution
Sets the resolution of all R&S RT-ZVC channels.

"On" The resolution of the analog channels is applied to R&S RT-ZVC channels. The signal is automatically interpolated or decimated to get the analog resolution.

"Off" The resolution of R&S RT-ZVC channels is set in a way so that the record length of the waveforms is minimum 1000 samples.

Remote command:
`ZVC:RESCoupled` on page 1368

5.7.1.2 Power calculation

Access: [App Cockpit]> "Analysis" > "Multi-channel probe" > "Power Cal." tab

State
Enables the corresponding channel of the probe. The number of available channels depend on the characteristics of your multi-channel power probe.

Remote command:
`ZVC:Z<m>:V<n>[:STATe]` on page 1375
`ZVC:Z<m>:I<n>[:STATe]` on page 1372

Measurement Impedance
Sets the impedance of the probe channel for power calculations and measurements.

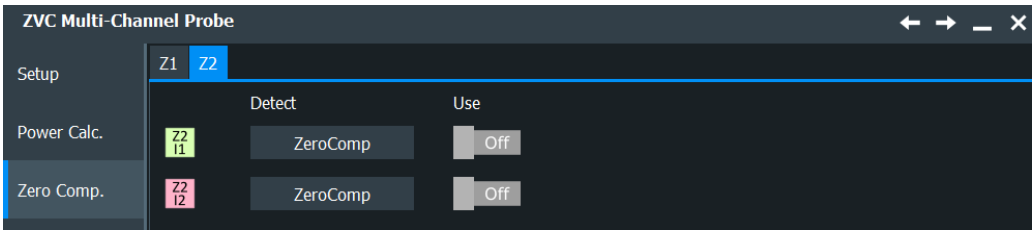
Remote command:
`ZVC:Z<m>:V<n>:IMPedance` on page 1373
`ZVC:Z<m>:I<n>:IMPedance` on page 1368

5.7.1.3 Zero compensation

Access: [App Cockpit]> "Analysis" > "Multi-channel probe"> "ZeroComp" tab

Zero compensation is used to avoid negative currents in measurement results, and to improve the measurement accuracy. The determined compensation is valid as long as the temperature is constant and the probe settings are unchanged.

1. Disconnect the DUT from the power supply.
2. Adjust the current settings, in particular, the shunt settings. See Chapter 5.7.3, "ZVC current settings", on page 177.
3. Connect one probe tip of the R&S RT-ZVC probe to the high voltage pin of the DUT. Disconnect the other probe tip from the DUT.
4. Open the "ZeroComp" settings: [App Cockpit]> "Analysis" > "Multi-channel probe"> "ZeroComp" tab
5. Tap "ZeroComp" to determine the zero offset.
6. Enable "Use" to compensate for the zero offset.



ZeroComp
Measures the zero offset, the mean value on a currentless DUT. If temperature changes, or if you change the shunt or other probe settings, repeat the measurement.

Remote command:
`ZVC:Z<m>:I<n>:ZERComp:DETect` on page 1376

Use
If enabled, the measured "ZeroComp" offset value is compensated automatically.

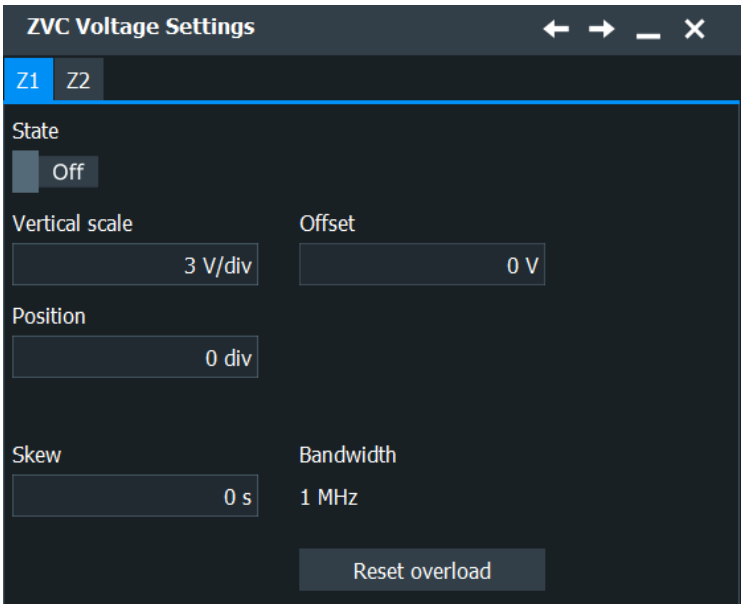
Remote command:
`ZVC:Z<m>:I<n>:ZERComp:USE` on page 1377

5.7.2 ZVC voltage settings

Access: [App Cockpit]> "Analysis" > "Multi-channel probe"> "Setup" tab > "Setup" icon of voltage channel

See also:

- Scale
- Offset



Channel
Selects the voltmeter channel to be configured.

State
Enables the corresponding voltage channel of the probe. The number of available channels depend on the characteristics of your multi-channel power probe.

Remote command:
`ZVC:Z<m>:V<n>[:STATe]` on page 1375

Position
Moves the selected signal up or down in the diagram. The visual effect is the same as for Offset. While the offset sets a voltage, position is a graphical setting given in divisions. Within a given operation range, modifying the position is equivalent to moving vertically the display range. The position can only be modified such that the display range reaches at most the limits of the operation range.

For the voltmeter channels, the Scale, the Offset and the "Position" specify the operating range of the voltmeter.

Remote command:
`ZVC:Z<m>:V<n>:POSition` on page 1374

Bandwidth
Displays the bandwidth of the current channel. You can set the probe bandwidth in the "Setup" dialog.

Remote command:
`ZVC:Z<m>:V<n>:BANDwidth?` on page 1373

Skew
Sets the skew, a delay value, that is known from the circuit specifics but cannot be compensated by the instrument automatically.

Remote command:
`ZVC:Z<m>:V<n>:SKEW` on page 1375

Reset Overload
Resets the overload indication at the probe.

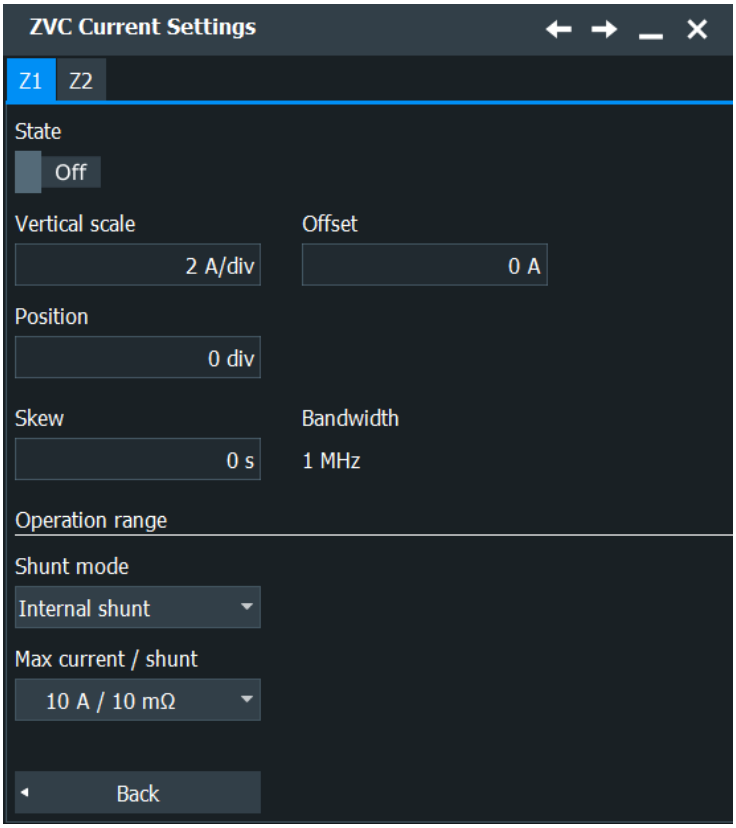
Remote command:
`ZVC:Z<m>:V<n>:OVERload:RSTO` on page 1374

5.7.3 ZVC current settings

Access: [App Cockpit]> "Analysis" > "Multi-channel probe"> "Setup" tab > "Setup" icon of current channel

See also:

- Scale
- Offset



Channel
Selects the amperemeter channel to be configured.

State
Enables the corresponding current channel of the probe. The number of available channels depend on the characteristics of your multi-channel power probe.

Remote command:
`ZVC:Z<m>:I<n>[:STATe]` on page 1372

Bandwidth
Displays the bandwidth of the current channel. You can set the probe bandwidth in the "Setup" dialog.

Remote command:
`ZVC:Z<m>:I<n>:BANDwidth?` on page 1367

Position
Moves the selected signal up or down in the diagram. The visual effect is the same as for Offset. While the offset sets a current, position is a graphical setting given in divisions. Within a given operation range, modifying the position is equivalent to moving vertically the display range. The position can only be modified such that the display range reaches at most the limits of the operation range.

Remote command:
`ZVC:Z<m>:I<n>:POSition` on page 1369

Skew
Sets the skew, a delay value, that is known from the circuit specifics but cannot be compensated by the instrument automatically.

Remote command:
`ZVC:Z<m>:I<n>:SKEW` on page 1372

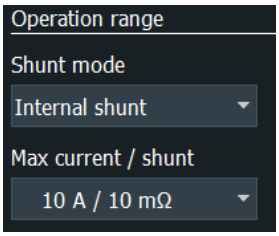
Shunt mode
Selects the internal or external shunt mode.

Regarding the shunt selection, i.e. the burden voltage level, there is a trade-off between the burden of the circuit under test and the SNR at the frontend input. The burden voltage is the DUT circuit loading caused by leads, connectors and the amperemeter circuit.

From the DUT perspective, the burden voltage has to be kept low not to distort the device operation. In contrast, from the probe's view the voltage has to be as large as possible to obtain a good SNR. For that reason, the external shunt can be applied to get the best compromise of both for a specific measurement range.

Remote command:
`ZVC:Z<m>:I<n>:SHUNt:MODE` on page 1370

Internal Shunt Mode
If Shunt mode is set to "Internal shunt", include the settings for the internal shunt mode.



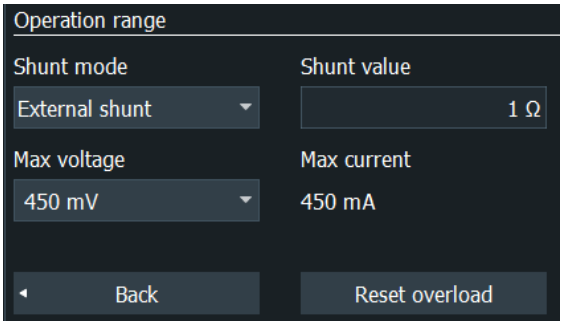
Max current / shunt ← Internal Shunt Mode
Selects the maximum current and the internal shunt value.

With the maximum current and the internal shunt selection, the operating range of the amperemeter is specified. At the same time, the burden voltage at the amperemeter input can be estimated. For values of the total round-trip resistance that can be seen at the test lead ends, consider the data sheet.

For using internal shunts, the circuit under test needs to be interrupted so that the current can flow through the probe.

Remote command:
`ZVC:Z<m>:I<n>:SHUNt:MAXCurrent` on page 1371

External shunt mode
If Shunt mode is set to "External shunt", include the settings for the external shunt mode.



Max voltage ← External shunt mode
Selects the maximum voltage at the external shunt.

Remote command:
`ZVC:Z<m>:I<n>:SHUNt:MAXVoltage` on page 1372

Shunt value ← External shunt mode
Sets the shunt value of the external shunt resistor.

Remote command:
`ZVC:Z<m>:I<n>:SHUNt:EVALue` on page 1370

Max current ← External shunt mode
Displays the maximum current.

Remote command:
`ZVC:Z<m>:I<n>:SHUNt:MXCValue?` on page 1371

Reset Overload ← External shunt mode
Resets the overload indication at the probe.

Remote command:
`ZVC:Z<m>:I<n>:OVERload:RSTO` on page 1369

5.8 Differential signals

With R&S RTO6, you can easily analyze differential signals using single-ended probes, or even cables. The instrument processes the input of single-ended probes on analog channels and creates the differential and common mode waveforms. Similar to Rohde & Schwarz modular probes, you can display the waveforms of differential voltage, common mode voltage, positive single-ended voltage and negative single-ended voltage. Triggering on differential signals is also possible.

Deembedding is possible if option R&S RTO6-K121 is installed.

Deembedding is applied to the two single-ended captured signals, resulting in deembedded differential waveform, common mode waveform, positive single-ended waveform and negative single-ended waveform. Deembedding also considers cross-couplings.

5.8.1 Settings of differential signals

Access: "Menu" > "Vertical" > "Differential"

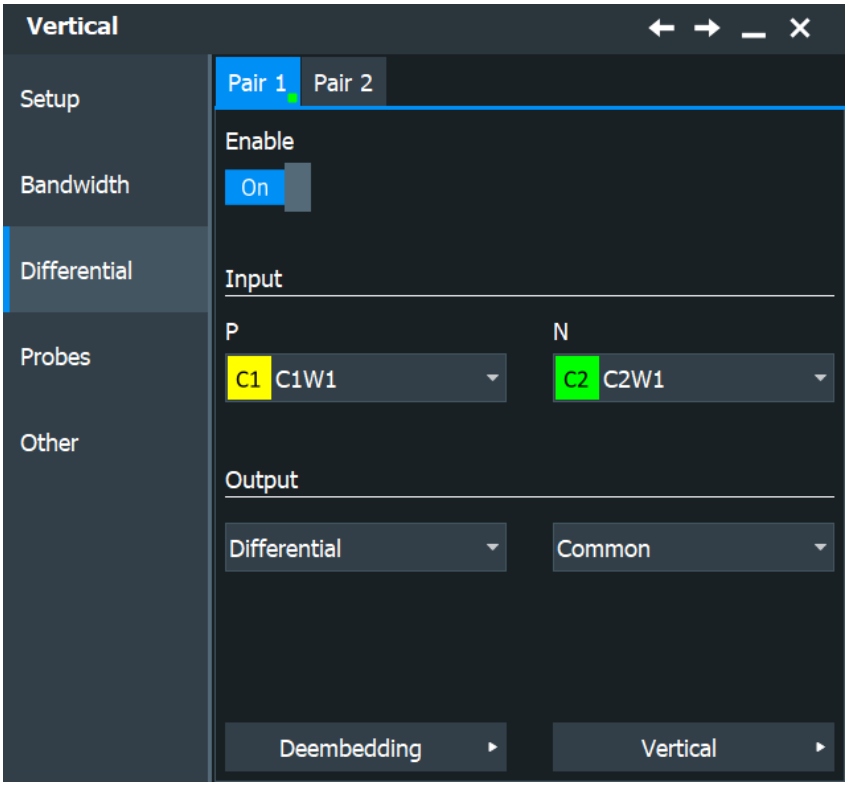


Figure 5-18: Setup of differential signals without deembedding

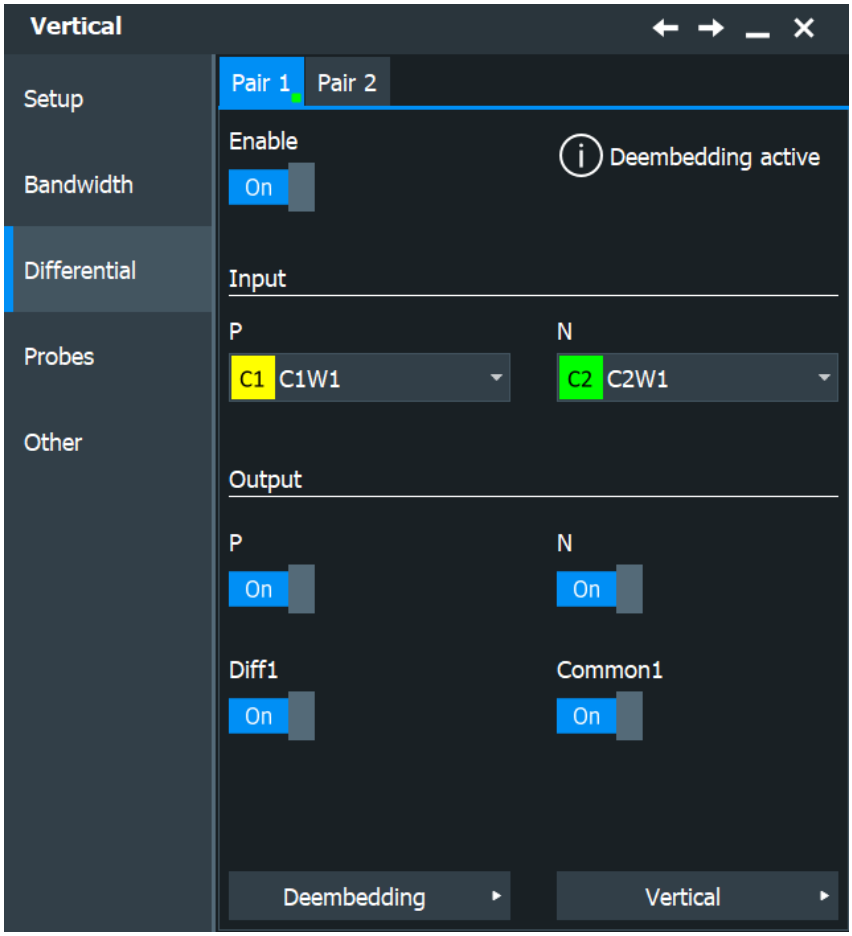


Figure 5-19: Setup of deembedded differential signals

Enable
Activates the differential signal.

Remote command:
`DIFFerential<m>:STATe` on page 1380

P, N
Select the analog input channel of the positive and negative signal.

Remote command:
`DIFFerential<m>:PSIGnal[:SELect]` on page 1380
`DIFFerential<m>:NSIGnal[:SELect]` on page 1380

Output: P, N, Diff, Common
Select the waveforms that are displayed as result of differential processing.

The number of output waveforms per differential signal depends on the deembedding:
- No deembedding: 2 output waveforms.
- With deembedding: up to 4 output waveforms. All output waveforms can be displayed at the same time.

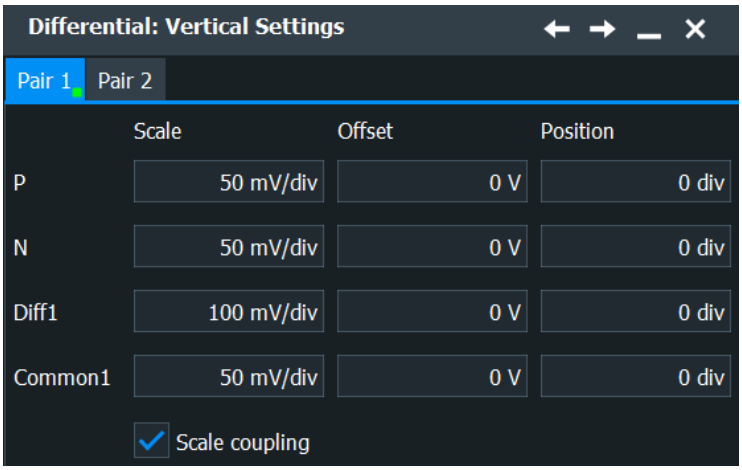
In the diagram, you see the output waveforms, and the signal icons of the input channels are grayed out. The input channels are only visible if "P" or "N" are selected as output.

"P", "N"	Displays the waveform of the positive or negative input signal. Without deembedding, the input channel is shown directly, no additional waveform is created. If differential deembedding is active, you always see the deembedded waveforms, but never the captured input signal.
"Differential"	Displays the differential signal as a new waveform.
"Common"	Displays the common mode signal as a new waveform.

Remote command:
`DIFFerential<m>:AOUTput` on page 1380
`DIFFerential<m>:BOUTput` on page 1380
If deembedding by software is active, use the following commands:
`DIFFerential<m>:COMMon:STATe` on page 1381
`DIFFerential<m>:DIFFerential:STATe` on page 1381
`DIFFerential<m>:NSIGnal:STATe` on page 1381
`DIFFerential<m>:PSIGnal:STATe` on page 1381

5.8.2 Vertical setup of differential signals

Access: "Menu" > "Vertical" > "Differential" > "Vertical Setup"



Scale
Sets the vertical scale of differential and common mode waveforms.

Vertical settings of P and N output are the vertical settings of the input channels.

Remote command:
`DIFFerential<m>:COMMon:SCALe` on page 1381
`DIFFerential<m>:DIFFerential:SCALe` on page 1381

Scale coupling
If enabled, the vertical scales of P, N, differential and common mode waveforms are coupled. The scales are related as follows:

$VertScale_P = VertScale_N = VertScale_{CM} = VertScale_{Diff}/2$

You can disable the scale coupling and set an individual scale for each waveform.

Remote command:
`DIFFerential<m>:COUPling` on page 1381

Offset
Sets the offset of differential and common mode waveforms.

Vertical settings of P and N output are the vertical settings of the input channels.

Remote command:
`DIFFerential<m>:COMMon:OFFSet` on page 1382
`DIFFerential<m>:DIFFerential:OFFSet` on page 1382

Position
Sets the vertical position of differential and common mode waveforms.

Vertical settings of P and N output are the vertical settings of the input channels.

Remote command:
`DIFFerential<m>:COMMon:POSition` on page 1382
`DIFFerential<m>:DIFFerential:POSition` on page 1382

5.8.3 Deembedding of differential signals

To remove the parasitic effects of the measurement setup from the measured signal, you can use deembedding for differential signals. Deembedding requires option R&S RTO6-K121.

The deembedding of differential signals is represented by one or more 4-port input components. You can activate deskewing, and set a skew time.

See also: Chapter 20, "Deembedding (option R&S RTO6-K121)", on page 1199.

1. On the "Vertical" menu, select "Differential Signals".
2. Select the input channels of the positive and negative signal.
3. Enable the differential signal.
4. Select "Deembedding".
5. Select the channel subtab of the positive input.
6. On the "Setup" tab, enable deembedding for the channel.
7. On the "Input" component, select "Configure".
8. If necessary, activate "Use skew offset". Use the deskewing if the P and N waveforms are not correctly aligned.

When you disable a deembedded differential signal in "Vertical" > "Differential Signals", deembedding is also disabled. When you enable the differential signal again, repeat the deembedding and output settings.

5.8.4 Analysis of differential signals

Before analyzing the output waveforms of differential signals, make sure that the P and N signals are correctly aligned. Without deembedding, use "Horizontal" > "Skew" to deskew the input waveforms if necessary. For deembedded waveforms, the skew setting is in the deembedding configuration.

For all output waveforms of differential signals (differential, common mode, P, N), the usual analysis methods are available:

- Trigger on differential, common mode, P, and N waveforms.
  If the differential signal is enabled ("Enable" = on), all possible output waveforms are available as trigger source, even if they are not selected and not displayed.
- Zoom
- Cursor measurements
- Automatic measurements
- Histogram
- Mathematics. In formulas, use DIFF1, DIFF2, COMMON1, COMMON2.
- FFT, spectrogram
- Mask test
- Export of waveform data. You can export the differential and common waveforms if they are selected as output and displayed. Raw data is not available.
  Without deembedding, P and N waveforms can be exported if they are selected as output and displayed (source = input channel). With software deembedding, only differential and common waveforms are available for export.

5.9 Setting up the waveform

This chapter contains the fundamental procedures for setting up the acquisition and adjusting the channel waveforms.

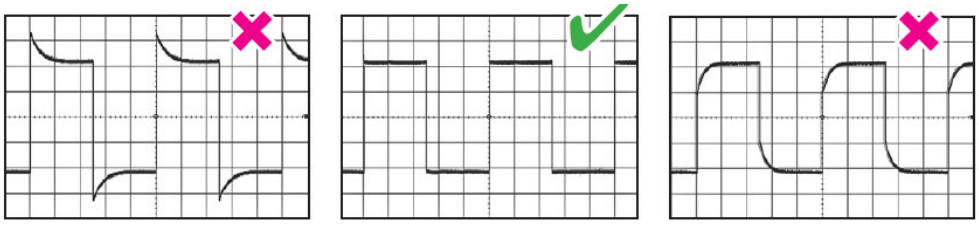
5.9.1 Adjusting passive probes

When using a passive probe, you have to compensate it when you connect it to the instrument the first time. Compensation matches the probe cable capacitance to the oscilloscope input capacitance to assure good amplitude accuracy from DC to upper bandwidth limit frequencies. A poorly compensated probe reduces the performance of the probe-oscilloscope system and introduces measurement errors resulting in distorted waveforms and inaccurate results.

Two connector pins are located on the front panel. The ⊥ pin is on ground level. The ⎍ pin supplies a square wave signal with 1 kHz for low frequency probe compensation.

1. Connect the BNC connector of the probe to one of the channel inputs.
2. Connect the probe's ground connector to the ground compensation pin, and the probe tip to the signal pin.
3. Press [Autoset].

   A square wave appears on the display.
4. Adjust the compensation trimmer of the probe to optimum square wave response. For details, refer to the documentation of your probe.



5.9.2 Setting up the signal input with autoset

Autoset is the solution for the major part of routine test-setup. It is also a good start if you need to use more complex trigger settings. Autoset finds appropriate horizontal and vertical scales, vertical offset, and trigger conditions to present a stable waveform.

1. Connect the probe to the input connector [C<n>].

   The instrument recognizes the probe and turns on the channel.
2. Press the [Autoset] key.

5.9.3 Adjusting the signal input manually

1. Connect the probe to the input connector [C<n>].

   The instrument recognizes the probe and turns on the channel.
2. Open "Menu" > "Horizontal" > "Setup" tab.
3. Set the "Time scale".
4. If you want to analyze the signal some time before or after the trigger, use the "Position" and "Reference point" to adjust the visible section of the waveform.
5. Press the channel button corresponding to the input channel. It is illuminated with the color of the channel waveform.
6. Open the "Menu" > "Vertical" > "Setup" tab.

7. Select the "Coupling".
8. Adjust the vertical "Scale", and the vertical "Position".
9. Proceed with the acquisition setup, see Chapter 5.9.4, "Setting the acquisition", on page 187.

5.9.4 Setting the acquisition

Prerequisites:

- Probes are connected.
- Vertical and horizontal settings are adjusted.

The settings are described in Chapter 5.4.2, "Mode settings", on page 129.

1. Open "Menu" > "Acquire" > "Mode" tab.
2. To configure the waveform-specific acquisition settings, enable "Multi waveform". Activate the waveforms "W1"/"W2"/"W3" you need.
   You can set up and display up to three waveforms per channel.
3. Select the "Decimation" - for example, Peak detect or High res.
4. Select the "Arithmetic" - for example, Average or Envelope.
   The instrument precludes incompatible combinations, like "Peak detect" with "Average".
5. If "Arithmetic"="Average" is selected for a waveform, enter the "Average count", that is the number of waveforms used for average calculation.
6. Set the aritmetic reset condition for the average and envelope calculation:
   a) If "Arithmetic Reset > Mode"="Time" is selected, enter the "Time".
   b) If "Arithmetic Reset > Mode"="Waveforms" is selected, enter the "Count".

5.9.5 Starting and stopping acquisition

You can control the acquisition in two ways:

- Running continuous acquisition until you stop it.
- Running one acquisition or a given number of acquisitions.
  If "Envelope" or "Average" is selected in the "Acquisition" tab, one acquisition means a cycle containing as many acquired waveforms as required to satisfy the reset conditions.

Prerequisites:

- Probes are connected.
- Vertical and horizontal settings are adjusted.
- Triggering is set.
- Channels to be acquired are turned on.

To start and stop continuous acquisition

1. Check if the trigger mode is set to "Norm". The trigger mode is shown in the trigger label in the upper right corner of the screen.
   If not, press the trigger [Auto Norm] key on the front panel to toggle the setting.
2. Press the [Run Stop] key to start acquisition.

   The acquisition starts if a trigger occurs.
3. To stop , press the [Run Stop] key again.

   The acquisition stops immediately.

To acquire a limited number of acquisitions

1. Check if the trigger mode is set to "Norm". The trigger mode is shown in the trigger label in the upper right corner of the screen.
   If not, press the trigger [Auto Norm] key on the front panel to toggle the setting.
2. Open "Menu" > "Acquire" > "Mode" tab.
3. Enter the number of acquisitions in the "Average count" field.
4. Press the [Single] key on the front panel.
   You can stop the running acquisition before it is finished by pressing the key again.

5.9.6 Using the roll mode

The roll mode can be used if the acquisition process is slow - that is if the time scale is large. In roll mode, the instrument shows the waveform immediately and saves waiting for the waveform display. The roll mode can be activated by the instrument if several conditions are fulfilled.

To set the roll mode manually

1. Make sure that all requirements for the roll mode are fulfilled: see Chapter 5.2, "Horizontal", on page 119.
2. Open "Menu" > "Horizontal" > "Roll" tab.
3. Set "Mode" to "Auto".
4. Set the "Start roll time", the acquisition time at which the instrument starts the roll mode.

5.9.7 Using fast segmentation

Fast Segmentation reduces the dead time between two waveform acquisition cycles.

The settings are described in Chapter 5.4.3, "Segmented settings", on page 132.

1. Open "Menu" > "Acquire" > "Segmented" tab.

2. Tap "Fast segmentation" to activate the fast segmentation mode.
3. If you want to sample the maximum number of acquisitions in a series, enable "Acquire maximum".
   If you want to capture a defined number of acquisitions, disable "Acquire maximum" and enter the "Required" number of acquisitions.
4. Set the "Display time".

5.9.8 Using digital filters

Before using digital filters, you determine if you want to filter input channels only or if the trigger signal is filtered too. The filter settings depend on this decision.

For details on filter settings and dependencies, see Chapter 5.5.5, "Digital filter setup", on page 143.

To filter the input channels only

1. Open "Menu" > "Vertical" > "Bandwidth" tab > "Digital Filter".
2. Set the "Trigger BW limit" to "Off".
3. Enter the "Channel BW limit" frequency for each filter.
4. Enable "Channel BW limit (On/Off)" for each channel to be filtered.

To filter the trigger signal

1. Open "Menu" > "Vertical" > "Bandwidth" tab > "Digital Filter".
2. Set the "Trigger BW limit" to "RF Reject".
3. Set the frequency limit for the filter: "Channel BW limit".
4. To filter the input channels too, enable "Channel BW limit (On/Off)" for each channel to be filtered.
   The trigger filter settings are applied also to these input channels.

6 Triggers

6.1 Basics of triggering

Triggering means to capture the interesting part of the relevant waveforms. Choosing the right trigger type and configuring all trigger settings correctly allows you to detect various incidents in analog, digital, and protocol signals.

Trigger

A trigger occurs if the complete set of trigger conditions is fulfilled. The trigger is the determining point in the waveform record. The instrument acquires continuously and keeps the sample points to fill the pre-trigger part of the waveform record. When the trigger occurs, the instrument continues acquisition until the post-trigger part of the waveform record is filled. Then it stops acquiring and waits for the next trigger. When a trigger is recognized, the instrument does not accept another trigger until the acquisition is complete and the holdoff time has expired.

Trigger setup

A simple trigger setup includes:

- Source of the trigger signal, its coupling and filtering
- Trigger type selection and setup
- Horizontal position of the trigger: see: Chapter 5.1.3.2, "Horizontal position", on page 115
- Trigger mode

The R&S RTO6 provides various trigger types for troubleshooting and signal analysis, for example, edge trigger, glitch trigger, interval trigger, pattern trigger, and much more.

For complex tasks like verifying and debugging designs, advanced trigger settings are available:

- Hysteresis, that is the rejection of noise to avoid unwanted trigger events caused by noise
- Holdoff to define exactly which trigger event causes the trigger
- Qualification to consider the states of digital signals on other input channels and their logical combination
- Trigger sequences to combine two trigger type conditions

Action on trigger

A trigger can initiate one or more actions, for example, saving a screenshot or saving waveform data. All available actions can be initiated at the same time.

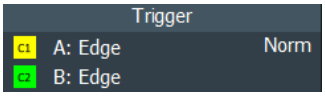
Trigger sequence

A complex trigger sequence joins two or more separate trigger conditions with an optional delay time and an optional reset time or reset condition. Similar setups are also known as multi-step trigger or A/B trigger.

Trigger information

Information on the most important trigger settings is shown in the trigger label on top of the signal bar. If you double-tap the trigger label, the "Trigger" dialog box opens. The label shows:

- Trigger mode
- Trigger type, edge/polarity and trigger source, for A- and B-trigger



6.2 Setting up the trigger

This chapter provides step-by-step procedures for the important stages of trigger setup.

6.2.1 Configuring a simple trigger

Prerequisites:

- Horizontal and vertical settings are set appropriately to the signals.
- The acquisition is running, the [Run Stop] key lights green.

For details on settings, see Chapter 6.3, "Trigger types", on page 193.

Proceed as follows:

1. Open the "Menu">"Trigger" dialog.

   The "Trigger" dialog box opens with the "Setup" tab.

2. Tap the "Source" button and select the trigger source.
3. Tap the "Type" button and select the trigger type.
4. Under "Trigger type dependent settings", configure the settings for the selected trigger type.
   See: Chapter 6.3, "Trigger types", on page 193
5. To set the trigger level automatically, tap "Find level".
6. Set the normal trigger mode. Do one of the following:
   - Press the [Auto Norm] key on the front panel until "Normal" is shown in the trigger label.

- In the "Mode / Holdoff" tab, set the "Trigger mode" to "Normal".

6.2.2 Positioning the trigger

By positioning the trigger on the time axis, you define which part of the waveform is displayed: mainly the pre-trigger part, or the post-trigger part, or the part around the trigger point.

For details on position settings, see Chapter 5.4, "Acquire settings", on page 126.

1. Open the "Menu"> "Horizontal" dialog.
2. Set the "Reference point" and the "Position".
   If you want to set the trigger position outside the waveform display, make sure that "Restrict horizontal position to acquisition range" is disabled.

6.2.3 Using holdoff

For details on holdoff settings, see Chapter 6.4, "Mode / Holdoff", on page 220.

1. Open the "Menu">"Trigger" dialog.
2. Select the "Mode / Holdoff" tab.
3. Select the "Holdoff mode".
4. Enter the "Holdoff settings" belonging to the selected mode.

6.2.4 Setting up an A → B → R trigger sequence

The complete configuration of a complex "A → B → R" trigger sequence consists of:

- A-trigger condition
- B-trigger condition in the same way as for the A-trigger, and optional delay time between the two triggers
- Optional reset by timeout and/or R-trigger

For details on sequence settings, see Chapter 6.8, "Sequence", on page 229.

1. Open the "Menu" > "Trigger" dialog.
2. Set "Trigger on" = "Sequence".
3. Select the type of "Sequence": "A → B → R".
4. Tap "Setup A trigger".
   A dialog opens where you can configure the first event.
   See: Chapter 6.2.1, "Configuring a simple trigger", on page 191.
5. Tap "Setup B trigger" and configure the B-trigger condition.

A dialog opens where you can configure the second event.

6. Optionally, set the "After 1st event wait" that the instrument waits after an A-trigger until it recognizes B-triggers.
7. Set the "2nd event count". The last B-trigger causes the trigger.
8. You can also define a reset condition. The sequence restarts with the A-trigger if no B-trigger occurs and the reset condition is fulfilled.
   a) To specify a reset by timeout, enable "Reset on timeout", and enter the "Rest timeout".
   b) To specify a reset trigger type condition, enable "Reset event" and configure the reset trigger type.
   The trigger types and settings depend on the A and B trigger settings. The instrument provides only possible, reasonable combinations.

6.2.5 Qualifying the trigger

Qualification considers the states of digital signals on other input channels and their logical combination as an additional trigger condition. For example, an edge trigger is configured for channel 1, and the instrument triggers only if the signal on channel 2 is high.

For details on qualification settings and restrictions, see Chapter 6.7, "Qualification", on page 226.

1. On the "Menu", select "Trigger" > "Qualify" tab.
2. Set the channel(s) with the input signal to be used as qualifying signal(s) to "On". The channel used as trigger source cannot be used for qualification.
3. Check and set the threshold levels for all used channels, that is, the thresholds for digitization of analog signals.
4. Set the signal state for each channel: high or low.
5. If more than one channel is selected, set the logical combination of the channel states.
6. If not yet enabled, tap "Qualify" to enable the qualification.

6.3 Trigger types

The setup of the trigger type is the most important part of the trigger definition. It determines the method to identify specific signal phenomena. Almost all trigger types are available for all conditions in a trigger sequence, that is, you can combine different types in the sequence. The instrument checks the trigger settings for compatibility and feasibility, and disables settings that do not fit the previous settings in the sequence.

The settings in the tab are:

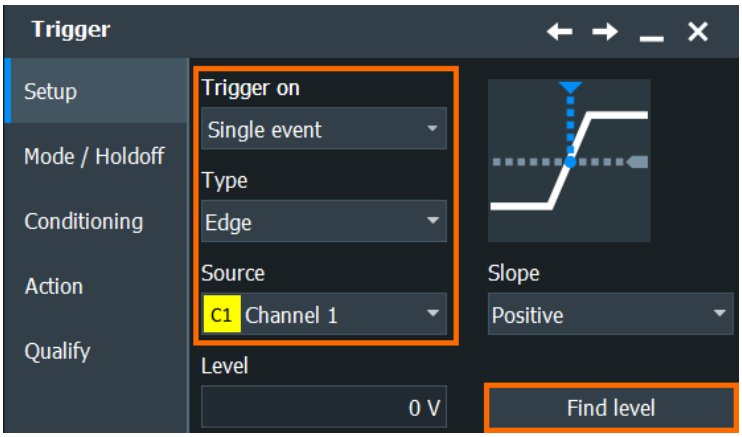
- Basic trigger settings....194
- Edge....196
- Glitch....197
- Width....198
- Runt....199
- Window....201
- Timeout....203
- Interval....204
- Slew rate....205
- Setup & Hold....207
- State....209
- Pattern....210
- Serial pattern....213
- TV/Video trigger....215
- NFC trigger....219
- CDR trigger....219
- Triggering on serial buses....219
- Triggering on parallel buses and digital channels....220

6.3.1 Basic trigger settings

Access: "Menu" > "Trigger" > "Setup" tab

The basic trigger settings are the trigger source and the trigger type, including the trigger level. These settings are specific for each condition in a trigger sequence. To set the trigger level automatically, use "Find level".

Depending on the trigger type, additional settings are available. They are explained in the trigger-type specific sections.



Trigger on
Selects, if you want to trigger on a single event, or on a series of events.

Remote command:
`TRIGger<m>:EVENt[:EVENt]` on page 1427

Source
Selects the source of the trigger signal for the current trigger condition. The trigger source works even if it is not displayed in a diagram. It must be synchronized to the signal to be displayed and analyzed.

The trigger source can be:
- Channel 1...4: An analog input channel
- Extern: External analog signal connected to the external trigger input
  The external trigger source is supported for the "Single event" sequence. It is not available if a longer trigger sequence is selected, or if qualification is enabled.
- Serial bus, D0...D15, Logic, Parallel bus 1...4:
  If options with trigger functionality are installed, the variety of trigger sources is enhanced with specific trigger sources.

Available sources depend on the trigger sequence setting. If "Single event" is selected, all inputs (analog input channels, serial and parallel buses, digital channels) can be used as trigger source. If any other trigger sequence is selected, only channel inputs Ch1...4 can be set as trigger source, and all other input sources are disabled.

See also: Chapter 6.8, "Sequence", on page 229

Remote command:
`TRIGger<m>:SOURce[:SELect]` on page 1389

Type
Selects the trigger type specific for each condition in a trigger sequence. The current trigger type is shown on the button.

The following trigger types are available:
- Edge, see page 196
- Glitch, see page 197
- Width, see page 198
- Runt, see page 199
- Window, see page 201
- Timeout, see page 203
- Interval, see page 204
- Slew rate, see page 205
- Setup & Hold, see page 207
- State, see page 209
- Pattern, see page 210
- Serial pattern, see page 213
- TV/Video trigger, see page 215
- CDR trigger, see page 1118 (requires option R&S RTO6-K13)

Restrictions:
- If the external trigger input is used as trigger source, the analog edge trigger is the only available trigger type.
- For the R-trigger (reset), the trigger types and settings are restricted dependent on the A and B trigger settings. The instrument provides only possible, reasonable combinations.

Remote command:
`TRIGger<m>:TYPE` on page 1390

Find level
Sets the trigger level automatically to *0.5 * (MaxPeak – MinPeak)*. The function is not available for an external trigger source and the TV trigger.

Remote command:
`TRIGger<m>:FINDlevel` on page 1392

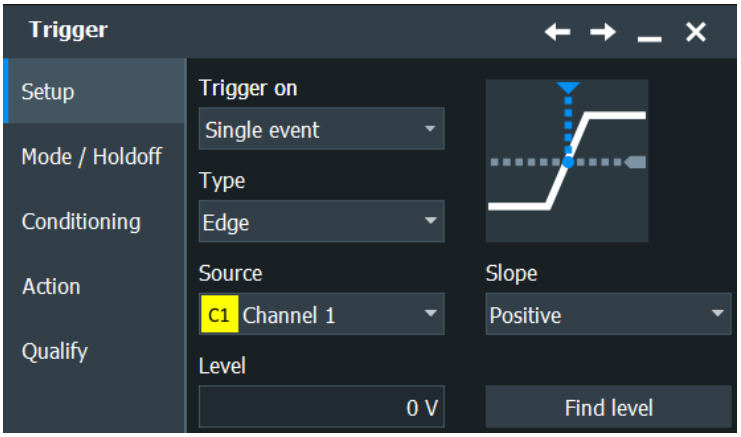
6.3.2 Edge

Access: "Menu" > "Trigger" > "Setup" tab > "Type = Edge"

The edge trigger is the most common trigger type. It is well known from analog oscilloscopes; and you can use it for analog and digital signals.

The trigger condition is fulfilled when the signal from the trigger source passes the specified threshold voltage in the specified direction (slope).

If the trigger source is a channel signal, the edge trigger uses the digitized trigger signal. This signal can be qualified and filtered with the DSP filter. If the trigger source is the external trigger input, the coupling and filter for this signal is set directly in the trigger setup.



Slope
Sets the edge type for the trigger condition.

"Positive"	Selects the rising edge, that is a positive voltage change.
"Negative"	Selects the falling edge, that is a negative voltage change.
"Both"	Selects the rising as well as the falling edge. This option is not available if the trigger source is the external trigger input.

Remote command:
`TRIGger<m>:EDGE:SLOPe` on page 1392
`TRIGger<m>:ANEDge:SLOPe` on page 1436
`TRIGger<m>:SLEW:SLOPe` on page 1403

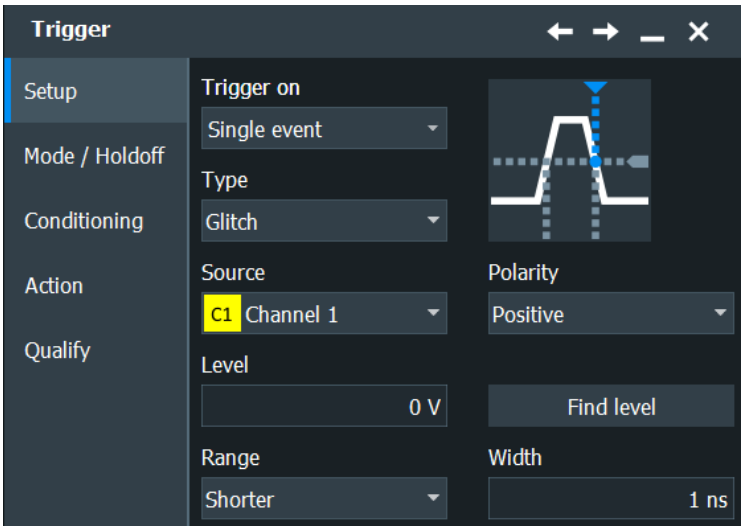
Level
Sets the voltage level for the trigger condition. You can also drag the trigger level marker on the display (TA or TB on the right edge of the display).

Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1391

6.3.3 Glitch

Access: "Menu" > "Trigger" > "Setup" tab > "Type = Glitch"

The glitch trigger detects pulses shorter or longer than a specified time. It identifies deviation from the nominal data rate and helps to analyze causes of even rare glitches and their effects on other signals.



Polarity
Indicates the polarity of a pulse, that is the direction of the first pulse slope.

"Positive" Selects positive going pulses.

"Negative" Selects negative going pulses.

"Either" Selects both positive and negative going pulses.

Remote command:
`TRIGger<m>:GLITch:POLarity` on page 1393
`TRIGger<m>:RUNT:POLarity` on page 1396

Range
Selects which glitches are identified: shorter or longer than the specified "Width".

Remote command:
`TRIGger<m>:GLITch:RANGe` on page 1393

Width
Sets the length of a glitch. The instrument triggers on pulses shorter or longer than this value. The minimum width is 100 ps.

You need to know the expected pulse widths of the circuit to set the glitch width correctly.

Remote command:
`TRIGger<m>:GLITch:WIDTh` on page 1394

Trigger level
Sets the voltage level for the trigger condition. You can also drag the trigger level marker on the display (TA or TB on the right edge of the display). The range of the trigger level is limited in a way so that always a hysteresis for stable trigger conditions is available.

Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1391

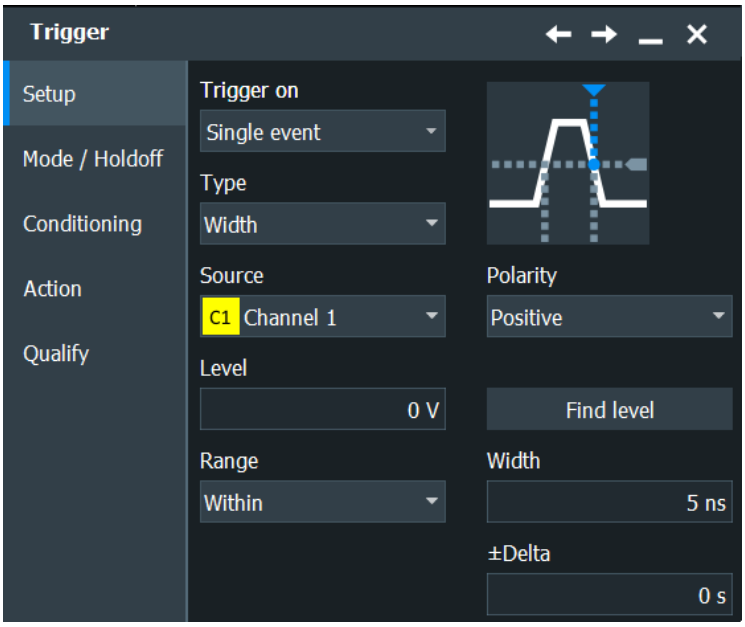
6.3.4 Width

Access: "Menu" > "Trigger" > "Setup" tab > "Type = Width"

The width trigger compares the pulse width (duration of a pulse) with a given time limit. It detects pulses with an exact pulse width, pulses shorter or longer than a given time, and pulses inside or outside the allowable time range. The pulse width is measured at the trigger level.

Using the width trigger, you can define the pulse width more precisely than with the glitch trigger. However, using the range settings "Shorter" and "Longer", you can also trigger on glitches.

The width trigger can only analyze **either** positive **or** negative polarity, but searching for a width is also possible for both polarities at the same time ("Either").



Polarity
Indicates the polarity of a pulse, that is the direction of the first pulse slope.

"Positive" Triggers on positive going pulses.

"Negative" Triggers on negative going pulses.

Remote command:
`TRIGger<m>:WIDTh:POLarity` on page 1394

Range
Selects how the range of a pulse width is defined:

"Within" Triggers on pulses inside a given range. The range of the pulse width is defined by "±Delta" related to "Width".

"Outside" Triggers on pulses outside a given range. The range definition is the same as for "Within" range.

"Shorter" Triggers on pulses shorter than the given "Width".

"Longer" Triggers on pulses longer than the given "Width".

Remote command:
`TRIGger<m>:WIDTh:RANGe` on page 1394

Width
For the ranges "Within" and "Outside", the width defines the center of a range which is defined by the limits ±Delta.

For the ranges "Shorter" and "Longer", the width defines the maximum and minimum pulse width, respectively.

Remote command:
`TRIGger<m>:WIDTh:WIDTh` on page 1395

±Delta
Defines a range around the given width value.

The combination "Range" = Within and "±Delta" = 0 triggers on pulses with a pulse width that equals "Width".

The combination "Range" = Outside and "±Delta" = 0 means to trigger on pulse widths ≠ "Width".

Trigger level
Sets the voltage level for the trigger condition. You can also drag the trigger level marker on the display (TA or TB on the right edge of the display). The range of the trigger level is limited in a way so that always a hysteresis for stable trigger conditions is available.

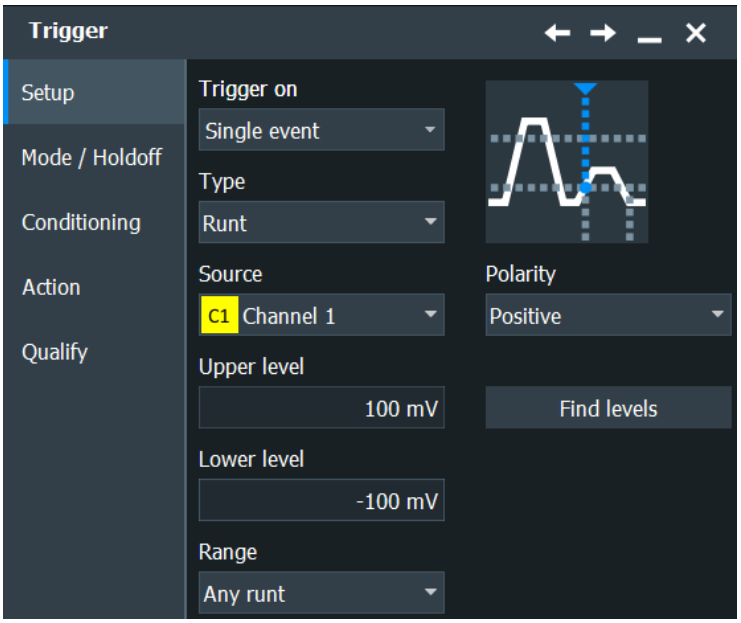
Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1391

6.3.5 Runt

Access: "Menu" > "Trigger" > "Setup" tab > "Type = Runt"

A runt is a pulse lower than normal in amplitude. The amplitude crosses the first threshold twice in succession without crossing the second one. In addition to the threshold amplitudes, you can define a time limit for the runt in the same way as for width triggers. For example, this trigger can detect logic, digital, and analog signals

remaining below a specified threshold amplitude because I/O ports are in undefined state.



Polarity
Indicates the polarity of a pulse, that is the direction of the first pulse slope.

"Positive" Selects positive going pulses.

"Negative" Selects negative going pulses.

"Either" Selects both positive and negative going pulses.

Remote command:
`TRIGger<m>:GLITch:POLarity` on page 1393
`TRIGger<m>:RUNT:POLarity` on page 1396

Upper level
Sets the upper voltage threshold.

Remote command:
`TRIGger<m>:LEVel<n>:RUNT:UPPer` on page 1396

Lower level
Sets the lower voltage threshold.

Remote command:
`TRIGger<m>:LEVel<n>:RUNT:LOWer` on page 1396

Range
Selects how the time limit of the runt pulse is defined:

"Any runt" Triggers on all runts fulfilling the level condition, without time limitation.

"Longer" Triggers on runts longer than the given "Runt width".

"Shorter" Triggers on runts shorter than the given "Runt width".

"Within" Triggers if the runt length is inside a given time range. The range is defined by "Runt width" and "±Delta".

"Outside" Triggers if the runt length is outside a given time range. The range definition is the same as for "Within" range.

Remote command:
`TRIGger<m>:RUNT:RANGe` on page 1396

Runt width
For the ranges "Shorter" and "Longer", the runt width defines the maximum and minimum pulse width, respectively.

For the ranges "Within" and "Outside", the runt width defines the center of a range which is defined by "±Delta".

Remote command:
`TRIGger<m>:RUNT:WIDTh` on page 1397

±Delta
Defines a range around the given runt width.

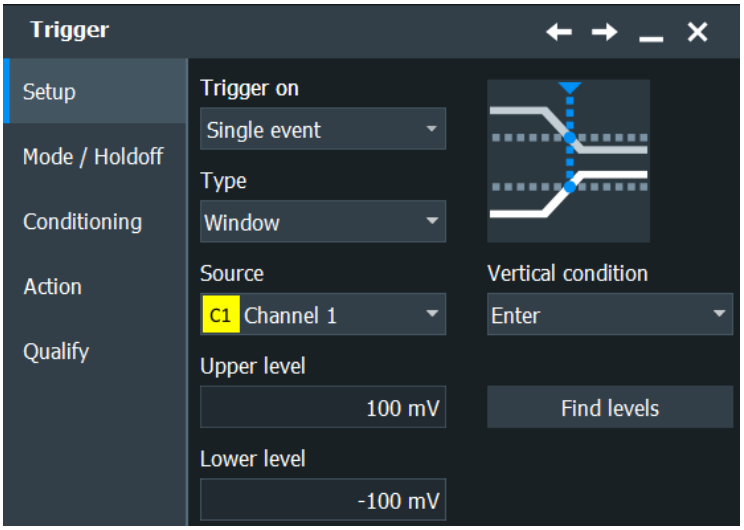
Remote command:
`TRIGger<m>:RUNT:DELTa` on page 1397

6.3.6 Window

Access: "Menu" > "Trigger" > "Setup" tab > "Type = Window"

The window trigger checks the signal run in relation to a "window". The window is formed by the upper and lower voltage levels. The trigger condition is fulfilled, if the waveform enters or leaves the window, or if the waveform stays inside or outside for a time longer or shorter than specified.

With the window trigger, you can display longer transient effects.



Vertical condition
Selects how the signal run is compared with the window:

"Enter"	Triggers when the signal crosses the upper or lower level and thus enters the window made up of these two levels.
"Exit"	Triggers when the signal leaves the window.
"Stay within"	Triggers if the signal stays between the upper and lower level for a specified time. The time is defined in various ways by the Time condition.
"Stay outside"	Triggers if the signal stays above the upper level or below the lower level for a specified time. The time is also defined by the "Time condition".

Remote command:
`TRIGger<m>:WINDow:RANGe` on page 1398

Upper level
Sets the upper voltage limit for the window.

Remote command:
`TRIGger<m>:LEVel<n>:WINDow:UPPer` on page 1398

Lower level
Sets the lower voltage limit for the window.

Remote command:
`TRIGger<m>:LEVel<n>:WINDow:LOWer` on page 1398

Time condition
Selects how the time limit of the window is defined. Time conditioning is available for the vertical conditions "Stay within" and "Stay outside".

"Within"	Triggers if the signal stays inside or outside the vertical window limits at least for the time *Width - Delta* and for *Width + Delta* at the most.
"Outside"	"Outside" is the opposite definition of "Within". The instrument triggers if the signal stays inside or outside the vertical window limits for a time shorter than *Width - Delta* or longer than *Width + Delta*.
"Shorter"	Triggers if the signal crosses vertical limits before the specified "Width" time is reached.
"Longer"	Triggers if the signal crosses vertical limits after the specified "Width" time is reached.

Remote command:
`TRIGger<m>:WINDow:TIME` on page 1399

Width
For the ranges "Within" and "Outside", the width defines the center of a time range which is defined by the limits "±Delta".

For the ranges "Shorter" and "Longer", it defines the maximum and minimum time lapse, respectively.

Remote command:
`TRIGger<m>:WINDow:WIDTh` on page 1400

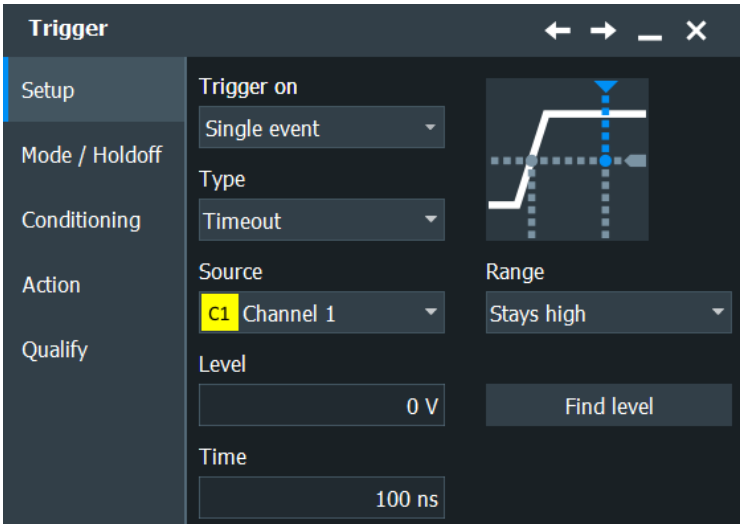
±Delta
Defines a range around the "Width" value.

Remote command:
`TRIGger<m>:WINDow:DELTa` on page 1400

6.3.7 Timeout

Access: "Menu" > "Trigger" > "Setup" tab > "Type = Timeout"

The timeout trigger checks if the signal stays above or below the threshold voltage for a specified time lapse. In other words, the trigger occurs if the trigger source does not have the expected transition within the specified time.



Trigger level
Sets the voltage level for the trigger condition. You can also drag the trigger level marker on the display (TA or TB on the right edge of the display). The range of the trigger level is limited in a way so that always a hysteresis for stable trigger conditions is available.

Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1391

Range
Selects the relation of the signal level to the trigger level:

"Stays high" The signal level stays above the trigger level.

"Stays low" The signal level stays below the trigger level.

"High or low" The signal level stays above or below the trigger level.

Remote command:
`TRIGger<m>:TIMeout:RANGe` on page 1400

Time
Defines the time limit for the timeout at which the instrument triggers.

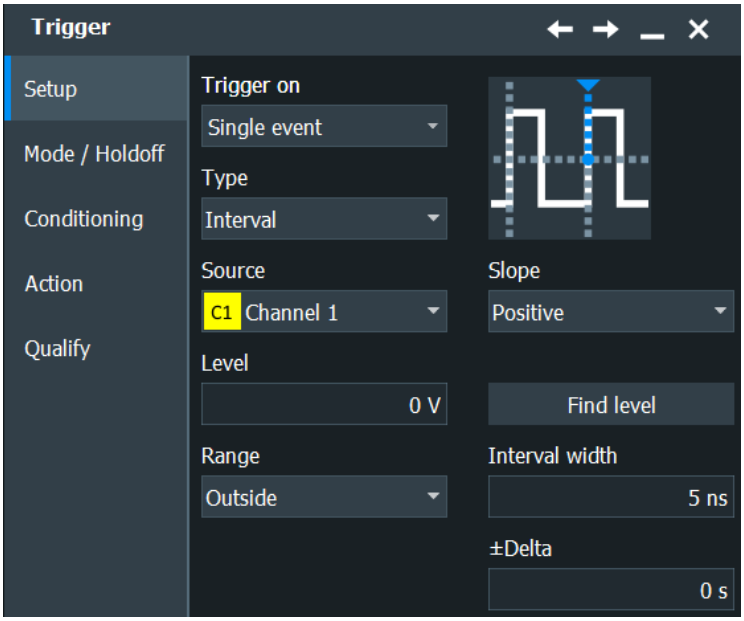
Remote command:
`TRIGger<m>:TIMeout:TIME` on page 1401

6.3.8 Interval

Access: "Menu" > "Trigger" > "Setup" tab > "Type = Interval"

The interval trigger analyzes the time between two pulses.

The interval trigger can analyze either rising or falling edges, but searching for an interval is also possible for both edges at the same time ("Either").



Slope
Sets the edge for the trigger. You can analyze the interval between positive edges or between negative edges.

Remote command:
`TRIGger<m>:INTerval:SLOPe` on page 1401

Trigger level
Sets the voltage level for the trigger condition. You can also drag the trigger level marker on the display (TA or TB on the right edge of the display). The range of the trigger level is limited in a way so that always a hysteresis for stable trigger conditions is available.

Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1391

Range
Selects how the range of an interval is defined:

"Within"	Triggers on pulse intervals inside a given range. The range is defined by "Interv. width" and "±Delta".
"Outside"	Triggers on intervals outside a given range. The range definition is the same as for "Within" range.
"Shorter"	Triggers on intervals shorter than the given "Interv. width".
"Longer"	Triggers on intervals longer than the given "Interv. width".

Remote command:
`TRIGger<m>:INTerval:RANGe` on page 1402

Interv. width
Defines the time between two pulses.

Remote command:
`TRIGger<m>:INTerval:WIDTh` on page 1402

±Delta
Defines a range around the "Interval width" value.

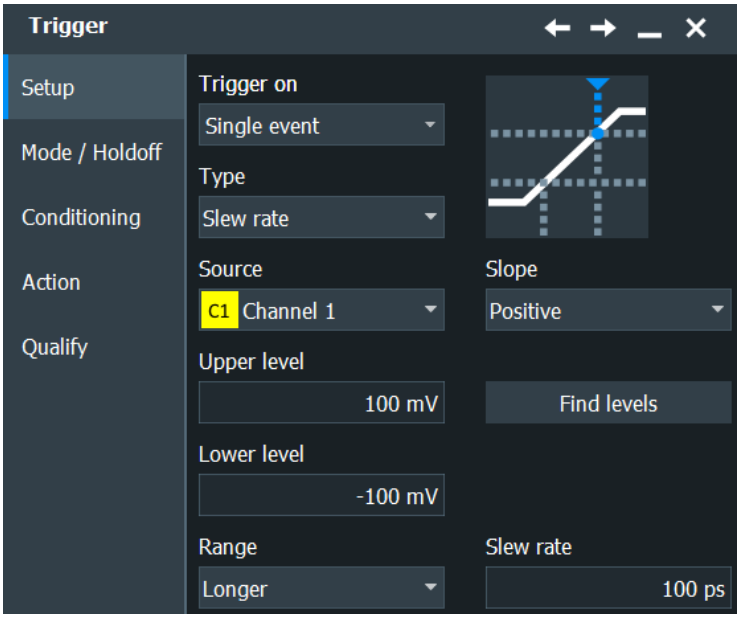
Remote command:
`TRIGger<m>:INTerval:DELTa` on page 1402

6.3.9 Slew rate

Access: "Menu" > "Trigger" > "Setup" tab > "Type = Slew rate"

The slew rate trigger is also known as transition trigger. It triggers if the transition time from the lower to higher voltage level (or vice versa) is shorter or longer as defined, or outside or inside a specified time range.

The slew rate trigger finds slew rates faster than expected or permissible to avoid overshooting and other interfering effects. It also detects slow edges violating the timing in pulse series.



Slope
Sets the edge type for the trigger condition.

"Positive"	Selects the rising edge, that is a positive voltage change.
"Negative"	Selects the falling edge, that is a negative voltage change.
"Both"	Selects the rising as well as the falling edge. This option is not available if the trigger source is the external trigger input.

Remote command:
`TRIGger<m>:EDGE:SLOPe` on page 1392
`TRIGger<m>:ANEDge:SLOPe` on page 1436
`TRIGger<m>:SLEW:SLOPe` on page 1403

Upper level
Sets the upper voltage threshold. When the signal crosses this level, the slew rate measurement starts or stops depending on the selected slope.

Remote command:
`TRIGger<m>:LEVel<n>:SLEW:UPPer` on page 1403

Lower level
Sets the lower voltage threshold. When the signal crosses this level, the slew rate measurement starts or stops depending on the selected slope.

Remote command:
`TRIGger<m>:LEVel<n>:SLEW:LOWer` on page 1403

Range
Selects how the time limit for the slew rate is defined. The time measurement starts when the signal crosses the first trigger level - the upper or lower level depending on the selected slope. The measurement stops when the signal crosses the second level.

"Within"	Triggers on slew rates inside a given time range. The range is defined by "Slew rate" and "±Delta".
"Outside"	Triggers on slew rates outside a given time range. The range definition is the same as for "Within" range.
"Shorter"	Triggers on slew rates shorter than the given "Slew rate" limit.
"Longer"	Triggers on slew rates longer than the given "Slew rate" limit.

Remote command:
`TRIGger<m>:SLEW:RANGe` on page 1404

Slew rate
For the ranges "Within" and "Outside", the slew rate defines the center of a range which is defined by the limits "±Delta".

For the ranges "Shorter" and "Longer", the slew rate defines the maximum and minimum slew rate limits, respectively.

Remote command:
`TRIGger<m>:SLEW:RATE` on page 1404

±Delta
Defines a time range around the given slew rate.

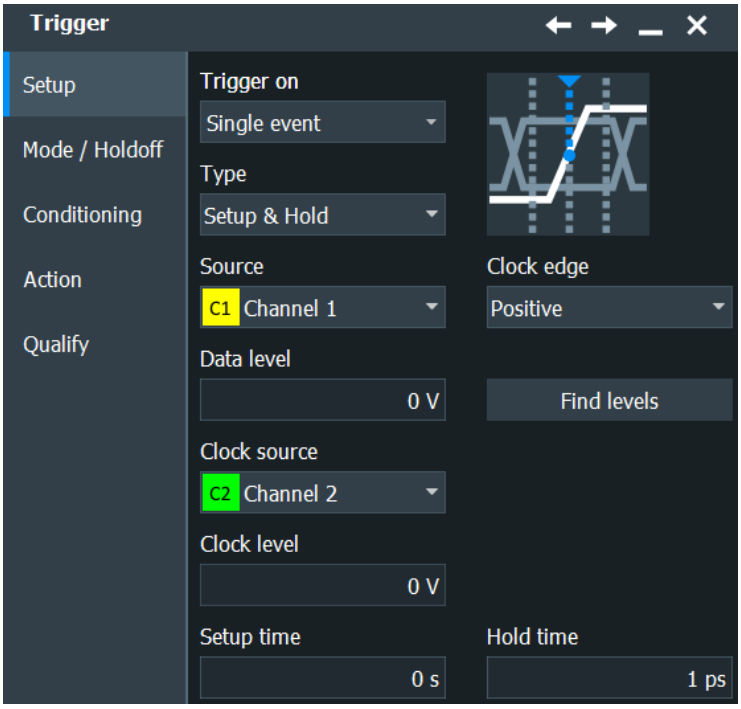
Remote command:
`TRIGger<m>:SLEW:DELTa` on page 1405

6.3.10 Setup & Hold

Access: "Menu" > "Trigger" > "Setup" tab > "Type = Setup & Hold"

With the Setup & Hold trigger you can analyze the relative timing between two signals: a data signal and the synchronous clock signal. Many systems require, that the data signal must be steady for some time before and after the clock edge, for example, the data transmission on parallel interfaces. With this trigger type, you can also test the time correlation of sideband and in-band signals.

The trigger occurs if the data signal crosses the data level during the setup and hold time. The reference point for the time measurement is defined by clock level and clock edge.



Clock source
Selects the input channel of the clock signal.

Remote command:
`TRIGger<m>:SETHold:CSOurce[:VALue]` on page 1406
`TRIGger<m>:SPATtern:CSOurce[:VALue]` on page 1411

Clock edge
Sets the edge of the clock signal to define the time reference point for the setup and hold time:

"Positive" Rising edge, a positive voltage change.

"Negative" Falling edge, a negative voltage change.

"Both" Both the rising and the falling edge.

Remote command:
`TRIGger<m>:SETHold:CSOurce:EDGE` on page 1405

Clock level
Sets the voltage level for the clock signal. Both "Clock level" and "Clock edge" define the starting point for calculation of the setup and hold time.

Remote command:
`TRIGger<m>:SETHold:CSOurce:LEVel` on page 1406

Data level
Sets the voltage level for the data signal. At this level, the setup and hold time is measured.

Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1391

Use data level for clock level
Sets the trigger levels and hysteresis values for all channels to the values of the currently selected trigger source. The function affects only the levels defined for the selected condition. The hysteresis of the external trigger input is an independent value, and it is not affected by level coupling.

Remote command:
`TRIGger<m>:SCOupling` on page 1407

Setup time
Sets the minimum time **before** the clock edge while the data signal must stay steady above or below the data level.

The setup time can be negative. In this case, the hold time is always positive. If you set a negative setup time, the hold time is adjusted by the instrument.

Remote command:
`TRIGger<m>:SETHold:STIMe` on page 1407

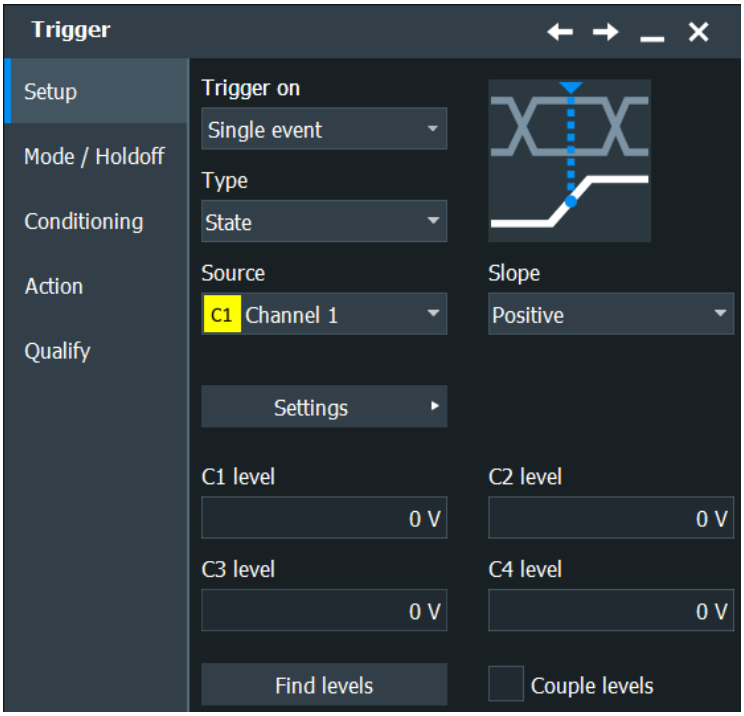
Hold time
Sets the minimum time **after** the clock edge while the data signal must stay steady above or below the data level.

The hold time can be negative. In this case, the setup time is always positive. If you set a negative hold time, the setup time is adjusted by the instrument.

Remote command:
`TRIGger<m>:SETHold:HTIMe` on page 1406

6.3.11 State

The state trigger is a qualified edge trigger. It combines the edge trigger settings with trigger qualification.



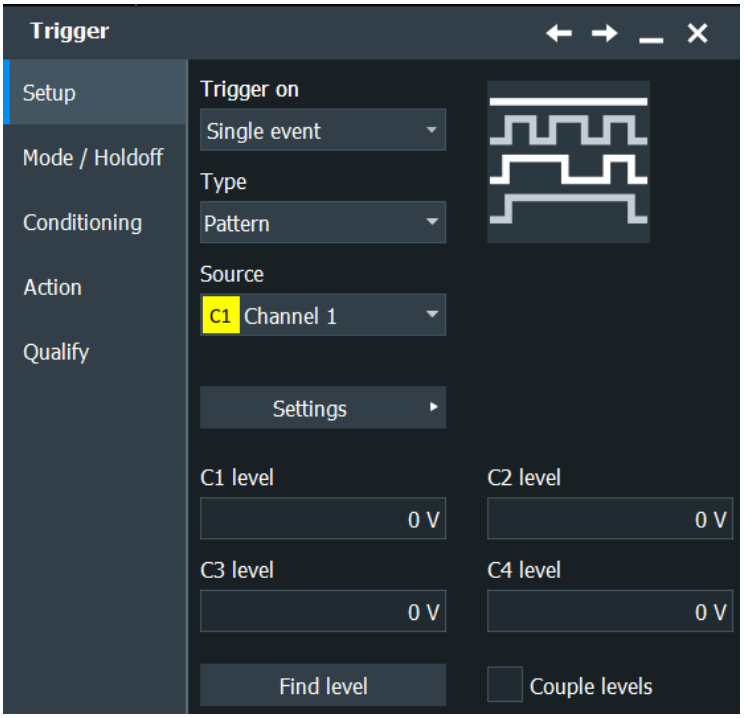
The individual settings are:

- "Slope" on page 196
- "Pattern" on page 211
- "Trigger Levels" on page 212
- "Find level" on page 196
- "Use data level for clock level" on page 209

6.3.12 Pattern

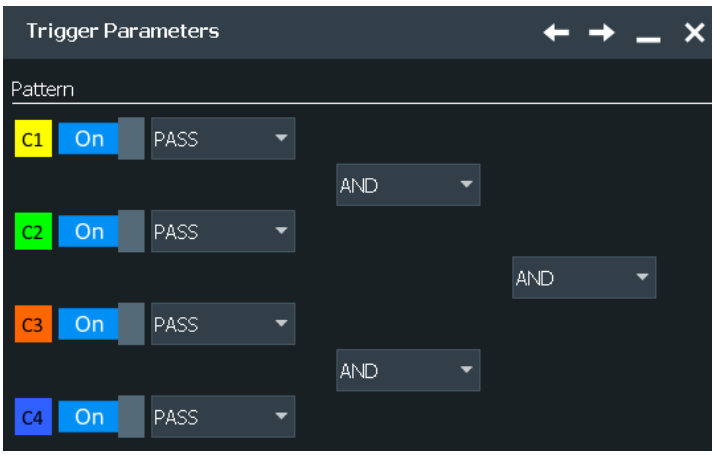
Access: "Menu" > "Trigger" > "Setup" tab > "Type = Pattern"

The pattern trigger is a logic trigger. It provides logical combination of the input channels and supports you in verifying the operation of digital logic. In addition to the pattern and the trigger levels, you can define a timing condition.



Pattern

The pattern contains the channel selection, and the logical operations structure of hardware-based Boolean logic.



"Channel"	Select the channels to be considered. For qualification, you can select all channel signals except for the trigger source. In pattern trigger setup, the trigger source channel is selected by default, and you can select all other channel signals.
"Boolean operator"	Defines the logical operation on the digital signal resulting from the comparison with the trigger level. • "PASS": leaves the input value unchanged • "NOT": inverts the input value

"Logical operator" Defines the logic combination of two sources. The sources are channel 1/2 and channel 3/4 on the first step, and in the second step the logical combination resulting from the first step.

- "AND": logical AND, conjunctive combination
- "NAND": logical NOT AND
- "OR": logical OR, disjunctive combination
- "NOR": logical NOT OR

Remote command:
`TRIGger<m>:QUALify<n>:A:LOGic` on page 1419
`TRIGger<m>:QUALify<n>:A[:ENABle]` on page 1418
`TRIGger<m>:QUALify<n>:AB:LOGic` on page 1420
`TRIGger<m>:QUALify<n>:ABCD:LOGic` on page 1420
`TRIGger<m>:QUALify<n>:B:LOGic` on page 1419
`TRIGger<m>:QUALify<n>:B[:ENABle]` on page 1418
`TRIGger<m>:QUALify<n>:C:LOGic` on page 1419
`TRIGger<m>:QUALify<n>:C[:ENABle]` on page 1418
`TRIGger<m>:QUALify<n>:CD:LOGic` on page 1420
`TRIGger<m>:QUALify<n>:D:LOGic` on page 1419
`TRIGger<m>:QUALify<n>:D[:ENABle]` on page 1418

Trigger Levels

Define the trigger level for each input channel. For state and pattern trigger, the trigger level is a decision threshold: If the signal value is higher than the trigger level, the signal state is high (1 or true for the Boolean logic). Otherwise, the signal state is considered low (0 or false) if the signal value is below the trigger level.

You can set the trigger levels for all channels to the same value, see "Use data level for clock level" on page 209.

Additional settings: Timing

"Timing" adds time limitation to the pattern condition.

"Off" No time limitation. The trigger occurs if the pattern condition is fulfilled.

"Timeout" Defines how long the result of the pattern condition must be true or false.

"Width" Defines a time range for keeping up the true result of the pattern condition. The range is defined in the same way as for width and interval triggers, see "Range" on page 199.

Remote command:
`TRIGger<m>:PATTern:MODE` on page 1408
`TRIGger<m>:PATTern:TIMeout:MODE` on page 1409
`TRIGger<m>:PATTern:TIMeout[:TIME]` on page 1409
`TRIGger<m>:PATTern:WIDTh:DELTa` on page 1410
`TRIGger<m>:PATTern:WIDTh:RANGe` on page 1409
`TRIGger<m>:PATTern:WIDTh[:WIDTh]` on page 1410

6.3.13 Serial pattern

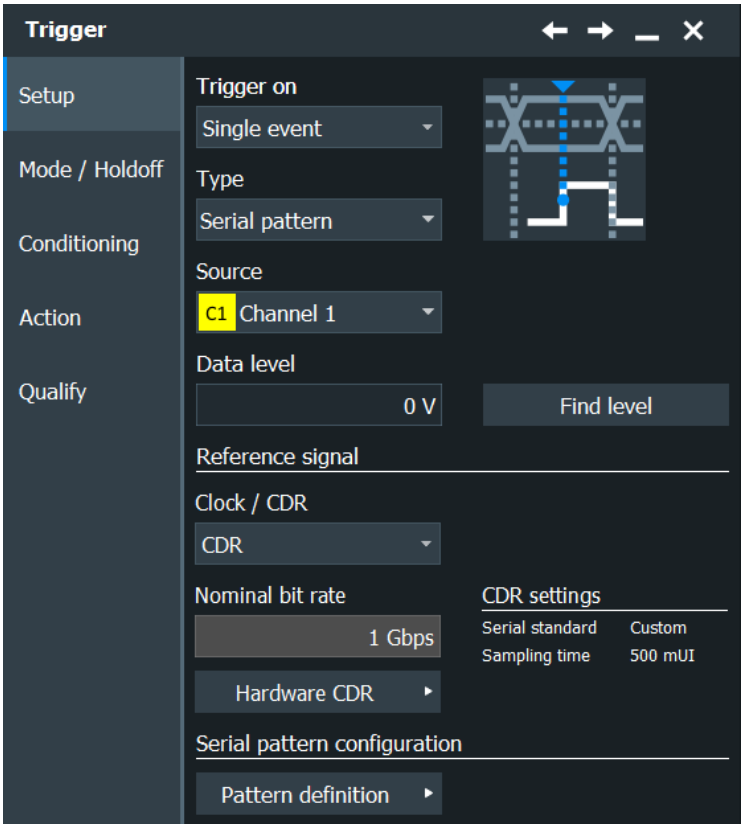
Access: "Menu" > "Trigger" > "Setup" tab > "Type = Serial Pattern"

The serial pattern is used to trigger on signals with serial data patterns in relation to a clock signal - for example, on bus signals like the I²C bus.

The instrument expects the bits coming in LSB first order. A triggered waveform in the diagram shows the LSB on the left and the MSB on the right side.

For convenient and comprehensive triggering on specific serial data, options for serial protocol analysis are provided.

If option R&S RTO6--K13 is installed, you can use the recovered clock signal for the serial pattern trigger instead of a real clock signal, see Chapter 18.3.2.3, "Serial pattern trigger using CDR", on page 1119.



Data level
Sets the voltage level for the data signal.

If the signal value is higher than the data level, the state is 1. Below the level, the signal state is 0.

Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1391

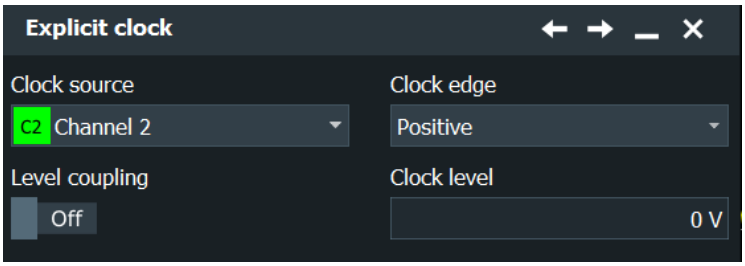
Clock / CDR
Selects if an explicit clock or a CDR signal is used as a reference signal for the serial pattern trigger.

"CDR" Uses clock data recovery. See Chapter 18.3.2.3, "Serial pattern trigger using CDR", on page 1119.

"Clock" Uses an explicit clock signal, see Explicit clock.

Remote command:
`TRIGger<m>:SPATtern:REFSource` on page 1412

Explicit clock
Opens a dialog to define the settings of the explicit clock settings.



Clock source ← Explicit clock
Selects the input channel of the clock signal.

Remote command:
`TRIGger<m>:SETHold:CSOurce[:VALue]` on page 1406
`TRIGger<m>:SPATtern:CSOurce[:VALue]` on page 1411

Clock edge ← Explicit clock
Together with the clock level, the clock edge sets the point in time when the state of the data signal is checked:

"Positive" Rising edge, a positive voltage change.

"Negative" Falling edge, a negative voltage change.

"Both" Both the rising and the falling edge.

Remote command:
`TRIGger<m>:SPATtern:CSOurce:EDGE` on page 1411

Clock level ← Explicit clock
Sets the voltage level for the clock signal.

Remote command:
`TRIGger<m>:SPATtern:CSOurce:LEVel` on page 1411

Use data level for clock level ← Explicit clock
Sets the trigger levels and hysteresis values for all channels to the values of the currently selected trigger source. The function affects only the levels defined for the selected condition. The hysteresis of the external trigger input is an independent value, and it is not affected by level coupling.

Remote command:
`TRIGger<m>:SCOupling` on page 1407

Pattern
The pattern contains the bits of the serial data to be found in the data stream. The maximum length of the pattern is 128 bit. Touch and hold the "Pattern" field to open the "Bit Pattern Editor" where you can enter the pattern in various formats.

See also: Chapter 13.1.7, "Bit pattern editor", on page 493.

In binary format, an *X* indicates that the logical level for the bit is not relevant (do not care).

Remote command:
`TRIGger<m>:SPATtern:PATTern` on page 1412

6.3.14 TV/Video trigger

Access: "Menu" > "Trigger" > "Setup" tab > "Type = TV"

The TV or video trigger is used to analyze analog baseband video signals. You can trigger on baseband video signals from standard definition and high definition standards, and also on user defined signals.

The instrument triggers on the line start - the horizontal sync pulse. You can trigger on all lines, or specify a line number. You can also trigger on the field or frame start.

Also, a delay can be set: Set the "Holdoff events" in the "Mode / Holdoff" tab to the number of fields to be skipped.

For details on holdoff settings, see Chapter 6.4, "Mode / Holdoff", on page 220.

Make sure that the trigger level crosses the synchronizing pulses of the video signal, see "Trigger level" on page 217.

Most video signals have an output impedance of 75 Ω. The channel inputs of the R&S RTO6 have an input impedance of 50 Ω or 1 MΩ. Make sure to provide the adequate matching to ensure amplitude fidelity. A simple 75 Ω feed-through termination combined with 1 MΩ oscilloscope inputs is suitable for most applications.

Once the trigger is set correctly, you can use cursor and automatic measurements to perform amplitude and timing measurements.

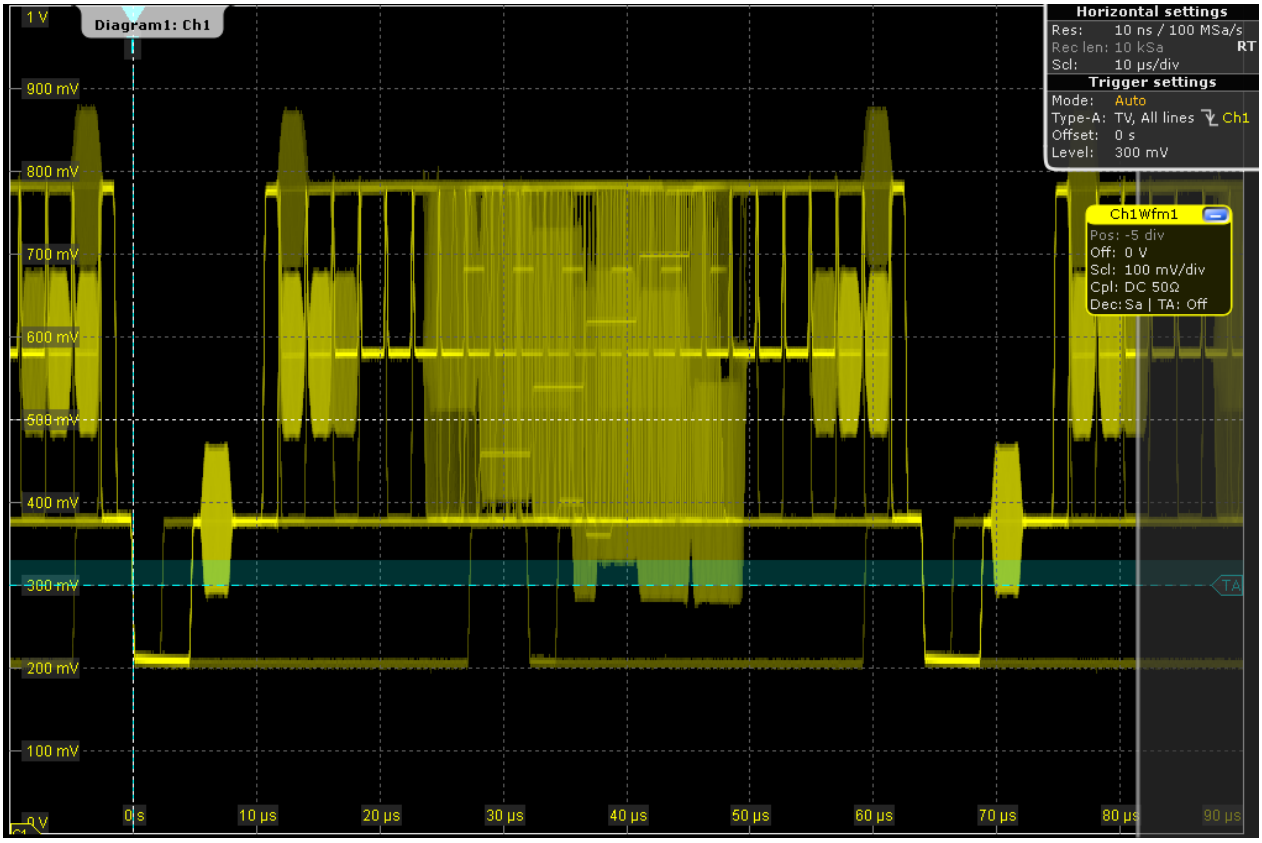


Figure 6-1: Trigger on all lines of a PAL signal with positive signal polarity, trigger level = 300 mV

Standard

Selects the TV standard or "Custom" for user-defined signals.

HDTV standards are indicated by the number of active lines, the scanning system (p for progressive scanning, i for interlaced scanning) and the frame rate. For interlaced scanning, the field rate is used instead of the frame rate. 1080p/24sF is an HDTV standard using progressive segmented frame scanning.

"Custom" can be used for signals of other video systems, for example, medical displays, video monitors, and security cameras. To trigger on these signals, you have to define the pulse type and length of the sync pulse, the scanning system and the line period.

Remote command:

`TRIGger<m>:TV:STANdard` on page 1413

Mode
Selects the lines or fields on which the instrument triggers. Available modes depend on the scanning system of the selected standard.

"All fields"	Triggers on the first video line of the frame (progressive scanning) or field (interlaced scanning), for example, to find amplitude differences between the fields.
"Odd fields / Even fields"	Triggers on the first video line of the odd or even field. These modes are available for interlaced scanning (PAL, PAL-M, SECAM, NTSC, 1080i) and progressive segmented frame scanning (1080p/24sF). They can be used, for example, to analyze the components of a video signal.
"All lines"	Triggers on the line start of all video lines, for example, to find maximum video levels.
"Line number"	Triggers on a specified line. Enter the line number in "Line #".

Remote command:
`TRIGger<m>:TV:MODE` on page 1413

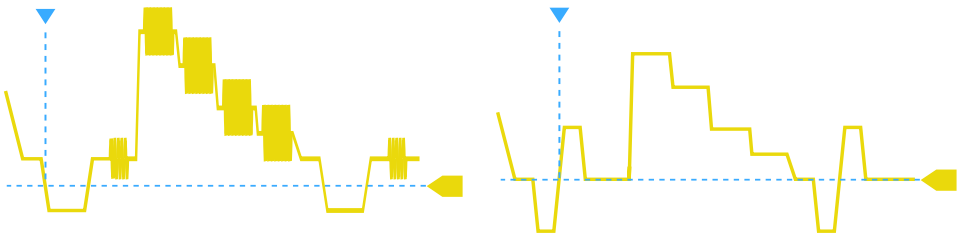
Line #
Sets the number of the line to be triggered on if "Mode" is set to "Line number". Usually the lines of the frame are counted, beginning from the frame start.

For NTSC signals, the lines are counted per field, not per frame. Therefore, you have to set the "Field" (odd or even), and the line number in the field.

Remote command:
`TRIGger<m>:TV:LINE` on page 1414
`TRIGger<m>:TV:LFIeld` on page 1415

Trigger level
Sets the trigger level as threshold for the sync pulse. Make sure that the trigger level crosses the synchronizing pulses of the video signal.

The hysteresis is set according to the settings in the "Noise Reject" tab.



Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1391

Signal polarity
Sets the polarity of the signal. Note that the sync pulse has the opposite polarity, for example, a positive signal has a negative sync pulse.

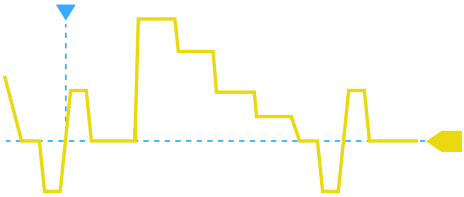


Figure 6-2: Signal with positive polarity and tri-level sync pulse

Remote command:
`TRIGger<m>:TV:POLarity` on page 1414

Pulse type
Sets the type of the sync pulse, either bi-level sync pulse (used in SDTV signals), or tri-level sync pulse (used in HDTV signals).

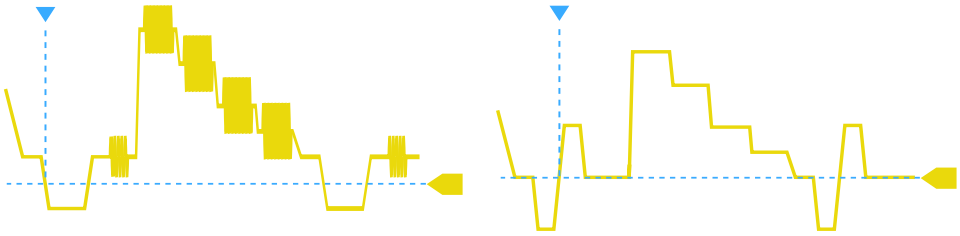
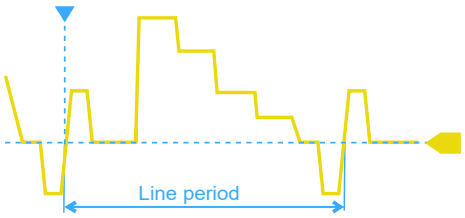


Figure 6-3: Bi-level (left) and tri-level (right) sync pulses

This setting is available for user-defined video signals if "Standard" is set to "Custom".

Remote command:
`TRIGger<m>:TV:CUSTom:STYPe` on page 1416

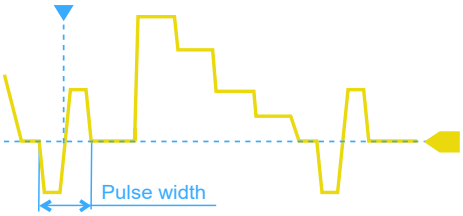
Line period
Sets the duration of a single video line, the time between two successive sync pulses.



This setting is available for user-defined video signals if "Standard" is set to "Custom".

Remote command:
`TRIGger<m>:TV:CUSTom:LDURation` on page 1416

Pulse width
Sets the width of the sync pulse.



This setting is available for user-defined video signals if "Standard" is set to "Custom".

Remote command:
`TRIGger<m>:TV:CUSTom:SDURation` on page 1417

Scan
Sets the scanning system.

This setting is available for user-defined video signals if "Standard" is set to "Custom".

"Interlaced" Interlace scanning uses two fields to create a frame. One field contains all the odd lines (odd, first, or upper filed), the other contains all the even lines of the image (even, second, or lower field). First the lines of the odd filed are processed, then the lines of the even field.

"Progressive" Progressive scanning is a method to capture, transmit and display all lines of a frame in sequence.

"Segmented" Progressive segmented frame uses progressive scanning to capture the frame, and interlaced scanning for transmission and display.

Remote command:
`TRIGger<m>:TV:CUSTom:SCANmode` on page 1415

6.3.15 NFC trigger

The Near Field Communication (NFC) trigger triggers on characteristic events of NFC signals. This trigger type requires option R&S RTO6--K11 "I/Q Software Interface".

For details, see Chapter 17.4, "NFC trigger", on page 1066

6.3.16 CDR trigger

The clock data recovery (CDR) trigger triggers on the edges of a clock edge stream that is recovered from a data signal using the hardware CDR. This trigger type requires option R&S RTO6--K13 "CDR".

For details, see Chapter 18.3.2.2, "CDR trigger", on page 1118.

6.3.17 Triggering on serial buses

Protocol analysis including configuration, triggering, and decoding is described in Chapter 13, "Protocol analysis", on page 484.

For information on triggering on serial buses, see the "Trigger" chapter of the relevant protocol.

6.3.18 Triggering on parallel buses and digital channels

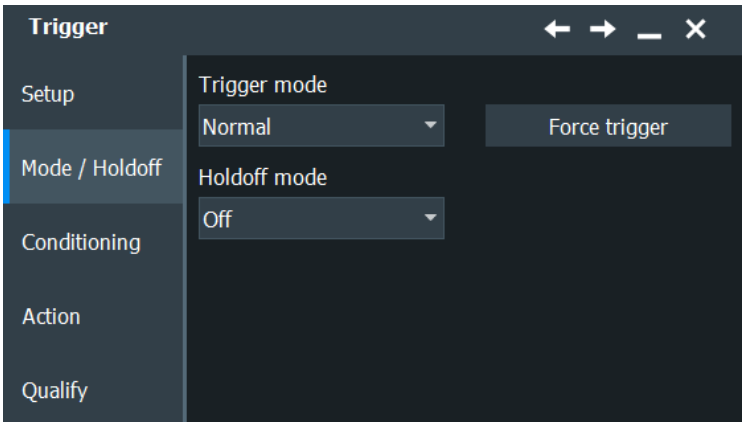
Triggering on digital signals requires the Mixed Signal Option. The option is described in Chapter 14, "Mixed signal option (MSO, R&S RTO6-B1)", on page 984.

For information on triggering, see Chapter 14.3.1, "Trigger settings for digital signals and parallel buses", on page 997.

6.4 Mode / Holdoff

Access: "Trigger" menu > "Mode / Holdoff" tab

Holdoff conditions define a waiting time after the current trigger until the next trigger can be recognized.



[Run Stop]/[Single]
Front panel keys to start and stop a continuous acquisition or a defined number of acquisition cycles, respectively. The number of acquisitions is set with "Average count".

Remote command:
`RUN` on page 1325
`SINGle` on page 1325
`STOP` on page 1326

Trigger mode
Sets the trigger mode which determines the behavior of the instrument if no trigger occurs. The current setting is shown on the trigger label on top of the signal bar.

To toggle quickly between "Auto" and "Normal" mode, use the [Auto Norm] key on the front panel (in "Trigger" section).

"Auto" The instrument triggers repeatedly after a time interval if the trigger conditions are not fulfilled. If a real trigger occurs, it takes precedence. This mode helps to see the waveform even before the trigger conditions are set correctly. The waveform on the screen is not synchronized, and successive waveforms are not triggered at the same point of the waveform. The time interval depends on the time base settings.

"Normal" The instrument acquires a waveform only if a trigger occurs, that is, if all trigger conditions are fulfilled. If no trigger occurs, no waveform is acquired and the last acquired waveform is displayed. If no waveform was captured before, none is displayed.
When no trigger has been found for longer than one second, a message box appears that shows the time elapsed since the last trigger.

"Repetitive" The instrument starts acquisition immediately and triggers after a short time interval independent of the time base settings and faster than in "Auto" mode. Real triggers are ignored. Use this mode if the "Auto" mode is too slow.

Remote command:
`TRIGger<m>:MODE` on page 1430

Force Trigger
If the acquisition is running in normal mode and no valid trigger occurs, forcing the trigger provokes an immediate single acquisition. Thus you can confirm that a signal is available and use the waveform display to determine how to trigger on it.

If you need this function frequently, you can add the "Force Trigger" icon to the toolbar.

Remote command:
`TRIGger<m>:FORCe` on page 1430

Holdoff mode
Selects the method to define the holdoff condition.

The trigger holdoff defines when the next trigger after the current will be recognized. Thus, it affects the next trigger to occur after the current one. Holdoff helps to obtain stable triggering when the oscilloscope is triggering on undesired events.

Holdoff settings are not available if the trigger source is an external trigger input or serial bus. For the TV trigger, only the "Events" mode is useful.

Example:

You want to analyze the first pulse in a burst of several pulses. At first, you select a sufficiently slow time base to display the entire burst. Then, you set the holdoff time a little longer than the length of the burst. Now, each trigger corresponds to the first pulse in successive bursts, and you can change the time base to display the waveform in more detail.

The following methods are available:

"Time" Defines the holdoff directly as a time period. The next trigger occurs only after the "Holdoff time" has passed.

"Events" Defines the holdoff as a number of trigger events. The next trigger only occurs when this number of events is reached. The number of triggers to be skipped is defined in "Holdoff events".

"Random" Defines the holdoff as a random time limited by "Minimum time" and "Maximum time". For each acquisition cycle, the instrument selects a new random holdoff time from the specified range.
Random holdoff prevents synchronization to discover effects invisible with synchronized triggering, for example, the features of a pulse train.

"Auto" The holdoff time is calculated automatically based on the current horizontal scale.
"Auto time scaling" defines the factor the horizontal scale is multipied with.
"Auto time" shows the resulting holdoff time: *Auto time = Auto time scaling * Horizontal scale*.

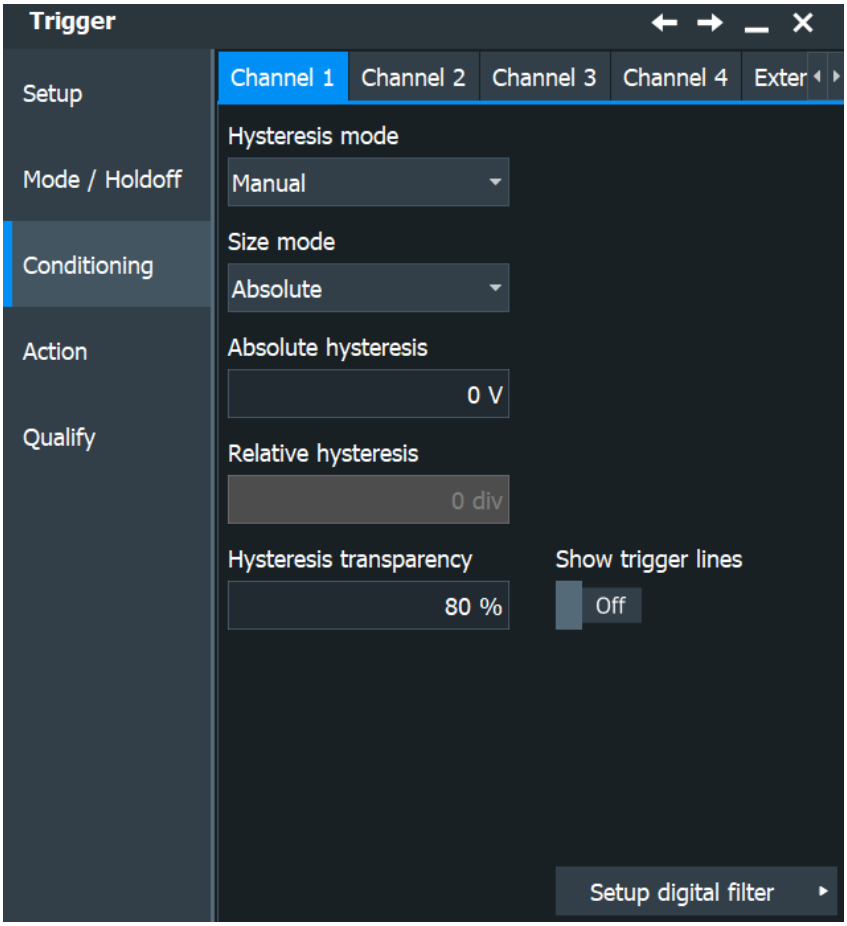
"Off" No holdoff

Remote command:
`TRIGger<m>:HOLDoff:MODE` on page 1420
`TRIGger<m>:HOLDoff:TIME` on page 1421
`TRIGger<m>:HOLDoff:EVENts` on page 1422
`TRIGger<m>:HOLDoff:MAX` on page 1423
`TRIGger<m>:HOLDoff:MIN` on page 1422
`TRIGger<m>:HOLDoff:AUTotime?` on page 1423
`TRIGger<m>:HOLDoff:SCALing` on page 1423

6.5 Conditioning

The rejection of noise by setting a hysteresis avoids unwanted trigger events caused by noise oscillation around the trigger level.

You can select the hysteresis mode and value for each channel separately, or couple the trigger levels and set the same hysteresis for channels. The hysteresis of the external trigger input is an independent value, and it is not affected by level coupling.



Hysteresis mode
Selects how the hysteresis is set.

"Auto" Automatic hysteresis is the recommended mode. The hysteresis is set by the instrument to reject the internal noise of the instrument.

"Manual" The hysteresis is defined directly in absolute or relative values.

Remote command:
`TRIGger<m>:LEVel<n>:NOISe[:STATe]` on page 1424

Size mode
Selects whether the hysteresis is defined in absolute or relative values. The setting is available only in manual hysteresis mode.

Remote command:
`TRIGger<m>:LEVel<n>:NOISe:MODE` on page 1425

Absolute hysteresis
Defines a range in absolute values around the trigger level. If the signal jitters inside this range and crosses the trigger level, no trigger event occurs.

Remote command:
`TRIGger<m>:LEVel<n>:NOISe:ABSolute` on page 1425

Relative hysteresis
Defines a range in divisions around the trigger level. If the signal oscillates inside this range and crosses the trigger level, no trigger event occurs.

Remote command:
`TRIGger<m>:LEVel<n>:NOISe:PERDivision` on page 1426
`TRIGger<m>:LEVel<n>:NOISe:RELative` on page 1426

Show trigger lines
Displays the trigger levels and the hysteresis in the diagrams until you disable this option.

Remote command:
`DISPlay:TRIGger:LINes` on page 1392

Hysteresis transparency
Defines the transparency of the hysteresis area above or below the trigger level.

Only visible if Show trigger lines is enabled.

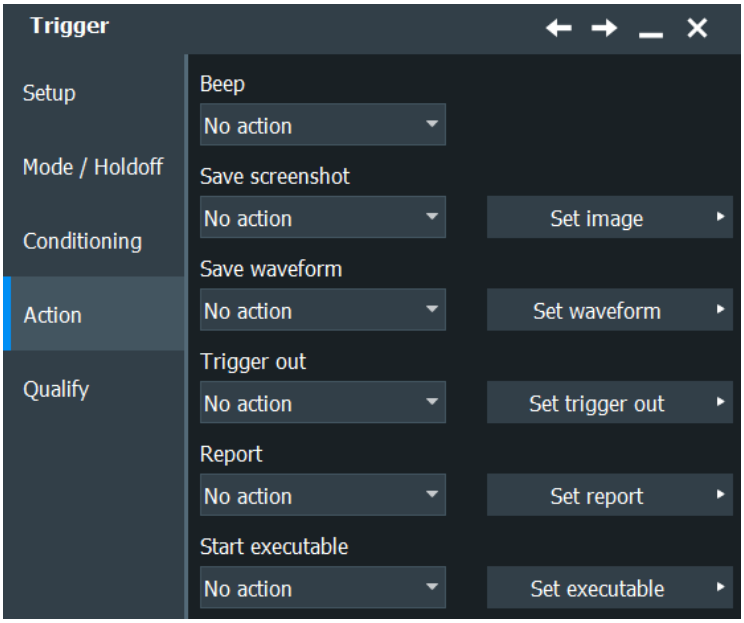
Hysteresis
Displays the automatic hysteresis value for "Hysteresis mode" = "Auto".

6.6 Actions on trigger

Access: "Menu" > "Trigger" > "Action" tab

The action settings define what happens when a trigger occurs. All available actions can be initiated at the same time.

The R&S RTO6 can provide an external trigger signal to synchronize the measurements of other instruments. The trigger out signal is also adjusted and enabled in the "Action" tab.



Actions on trigger

The trigger can initiate several actions, each time a trigger occurs. To activate an action, set it to "On trigger". The following actions are available:

"Beep"	Generates a beep sound.
"Save screenshot"	Saves a screenshot according to settings in "Menu" > "Save/Recall" > "Save" tab > "Screenshot".
"Save Waveform"	Saves the waveform data to a file according to settings in "Menu" > "Save/Recall" > "Save" tab > "Waveforms".
"Trigger out"	Selects, if a pulse is provided to the [Trigger Out] connector on the rear panel. Adjust the pulse with "Set trigger out", see "Trigger out signal setup" on page 225.
"Report"	Creates and saves a report using the settings defined in "Menu" key > "Save/Recall" key > "Save" tab > "Report".
"Start executable"	Starts an external application. Tap "Set executable" to set the application path and parameters. See: Chapter 4.8.3, "External application", on page 107.

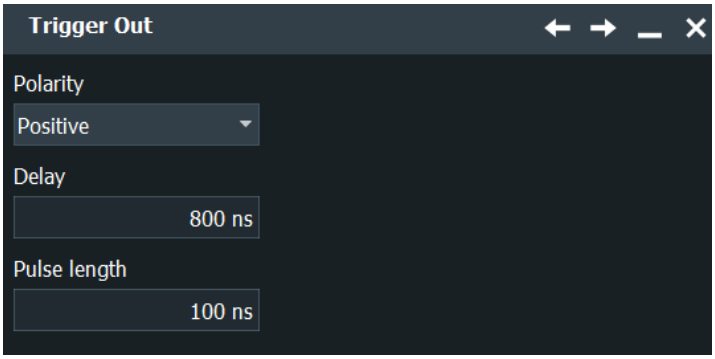
Remote command:

`TRIGger<m>:EVENt:BEEP` on page 1431
`TRIGger<m>:EVENt:PRINt` on page 1432
`TRIGger<m>:EVENt:WFMSave` on page 1432
`TRIGger<m>:EVENt:RUNexec` on page 1432
`TRIGger<m>:OUT:ACTion` on page 1433

Trigger out signal setup

Defines the pulse that is provided to the [Trigger Out] connector on the rear panel.

A trigger out pulse can be provided either when a trigger occurs, or when a mask test violation occurs, or when a limit check violation in a measurement occurs.



"Polarity" Sets the polarity of the trigger out pulse, that is the direction of the first pulse edge.

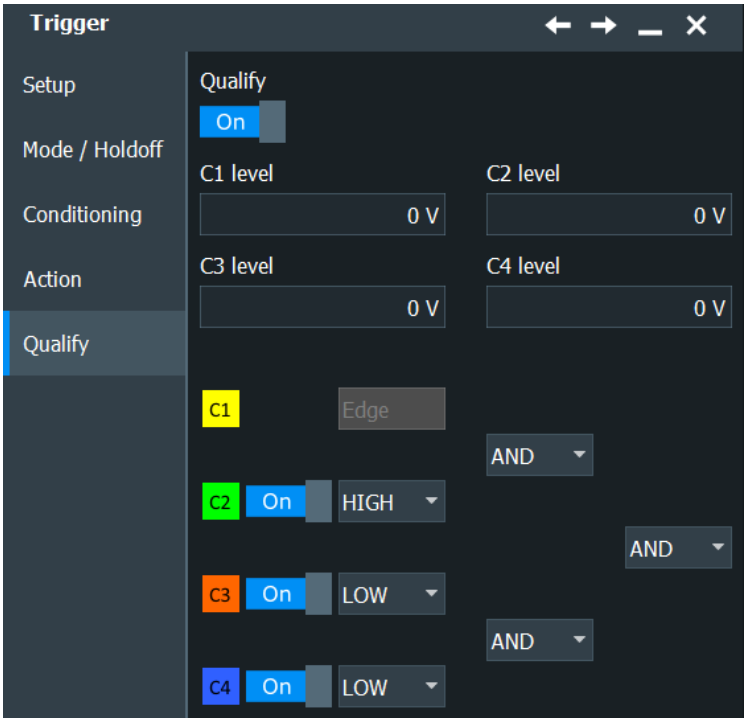
"Delay" Only available if "Trigger out" is set to "On trigger". The setting defines the delay of the first pulse edge to the trigger point. The minimum delay is 800 ns.

"Pulse length" Sets the length of the trigger out pulse.

Remote command:
`TRIGger<m>:OUT:POLarity` on page 1430
`TRIGger<m>:OUT:PLENgth` on page 1431
`TRIGger<m>:OUT:DELay` on page 1431

6.7 Qualification

Access: "Menu" > "Trigger" > "Qualify" tab



By qualifying a trigger event, you can logically combine the trigger signal with the state of other analog channel signals.

The instrument triggers if both of the following apply:

- The basic conditions of the trigger event definition are fulfilled.
- The logical conditions of the trigger qualification are true.

Qualification is only available for the A-event.

Qualification is not supported if:

- The trigger source is "Extern".
- One of the following trigger types is selected: slew rate, Setup & Hold, Serial pattern, TV, and NFC.

► To enable the qualification settings, select Qualify.

Example: Trigger on write access of a specific device of a bus system

In circuits using SPI, several slave devices use the same lines for reading and writing data, and each slave has its own select line. To trigger on write access of specific slave, the write line is the trigger source and the select line of the slave is set as qualifiying condition.

Qualify

Enables the settings for trigger qualification that are defined in the "Qualification" tab. Qualification adds additional trigger conditions considering the logic states of other digital channel signals.

The checkmark is only active if at least one qualification channel is selected.

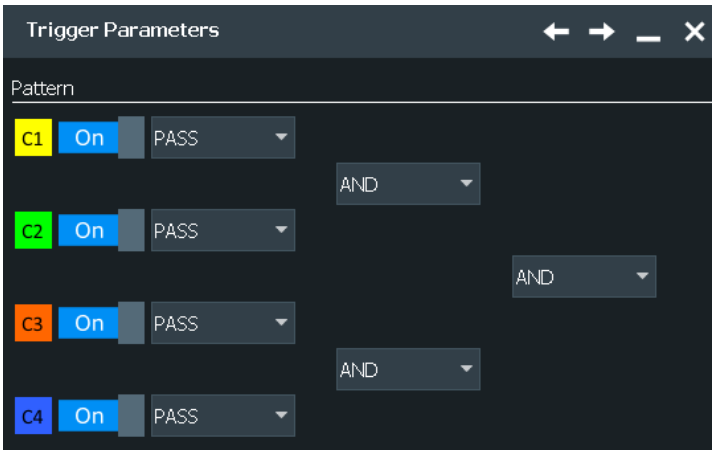
Qualification is available for many trigger types: Edge, Glitch, Width, Runt, Window, Timeout, and Interval.

Qualification is not possible for the R-event.

See also: Chapter 6.7, "Qualification", on page 226

Pattern

The pattern contains the channel selection, and the logical operations structure of hardware-based Boolean logic.



"Channel" Select the channels to be considered. For qualification, you can select all channel signals except for the trigger source. In pattern trigger setup, the trigger source channel is selected by default, and you can select all other channel signals.

"Boolean operator" Defines the logical operation on the digital signal resulting from the comparison with the trigger level.

- "PASS": leaves the input value unchanged
- "NOT": inverts the input value

"Logical operator" Defines the logic combination of two sources. The sources are channel 1/2 and channel 3/4 on the first step, and in the second step the logical combination resulting from the first step.

- "AND": logical AND, conjunctive combination
- "NAND": logical NOT AND
- "OR": logical OR, disjunctive combination
- "NOR": logical NOT OR

Remote command:
`TRIGger<m>:QUALify<n>:A:LOGic` on page 1419
`TRIGger<m>:QUALify<n>:A[:ENABle]` on page 1418
`TRIGger<m>:QUALify<n>:AB:LOGic` on page 1420
`TRIGger<m>:QUALify<n>:ABCD:LOGic` on page 1420
`TRIGger<m>:QUALify<n>:B:LOGic` on page 1419
`TRIGger<m>:QUALify<n>:B[:ENABle]` on page 1418
`TRIGger<m>:QUALify<n>:C:LOGic` on page 1419
`TRIGger<m>:QUALify<n>:C[:ENABle]` on page 1418
`TRIGger<m>:QUALify<n>:CD:LOGic` on page 1420
`TRIGger<m>:QUALify<n>:D:LOGic` on page 1419
`TRIGger<m>:QUALify<n>:D[:ENABle]` on page 1418

Trigger Levels

Define the trigger level for each input channel. For state and pattern trigger, the trigger level is a decision threshold: If the signal value is higher than the trigger level, the signal state is high (1 or true for the Boolean logic). Otherwise, the signal state is considered low (0 or false) if the signal value is below the trigger level.

You can set the trigger levels for all channels to the same value, see "Use data level for clock level" on page 209.

Use data level for clock level

Sets the trigger levels and hysteresis values for all channels to the values of the currently selected trigger source. The function affects only the levels defined for the selected condition. The hysteresis of the external trigger input is an independent value, and it is not affected by level coupling.

Remote command:
`TRIGger<m>:SCOupling` on page 1407

6.8 Sequence

A trigger sequence consists of at least one trigger condition and additional conditions defining when the trigger occurs.

The trigger sequence "A → B → R", for example, consists of two subsequent conditions: A-trigger and B-trigger with optional B-trigger delay and count. In addition, a reset condition R can be configured: timeout or R-trigger condition. A-, B-, and R-triggers are configured in the same way.

The instrument checks all trigger settings for compatibility and disables settings that do not fit the previous settings in the sequence.

After the A-trigger conditions have been met, and an optional delay has passed, the B-trigger with independent conditions is enabled. The instrument waits until one or a specified number of B-trigger conditions occur. If the reset condition is not fulfilled, the latest B-trigger causes the trigger event, and then the sequence starts again. The B-trigger can only cause the trigger event if it occurs after the A-trigger and after the delay time.

If you expect, for example, an irregular B-trigger, you can configure a reset condition to restart the sequence. The reset condition can be a simple timeout, and/or a reset event that is defined in the same way as the A- and B-trigger conditions.

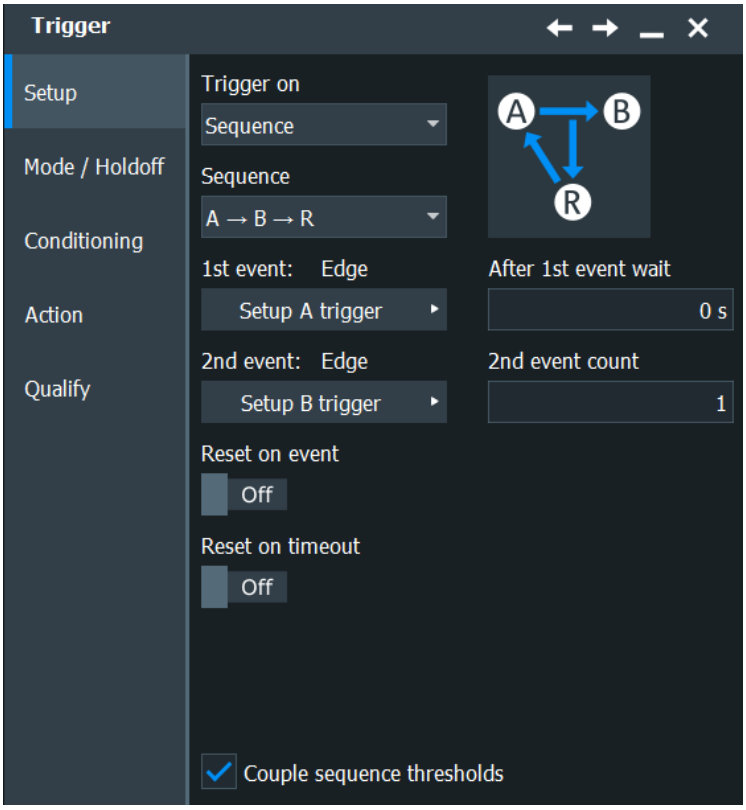
All trigger sequences require that input channels CH1...4 are set as trigger sources for all conditions. All other input sources are disabled. Trigger sequences are not available if one the following trigger types is set as A-trigger:

- Setup & Hold
- TV
- NFC
- CDR

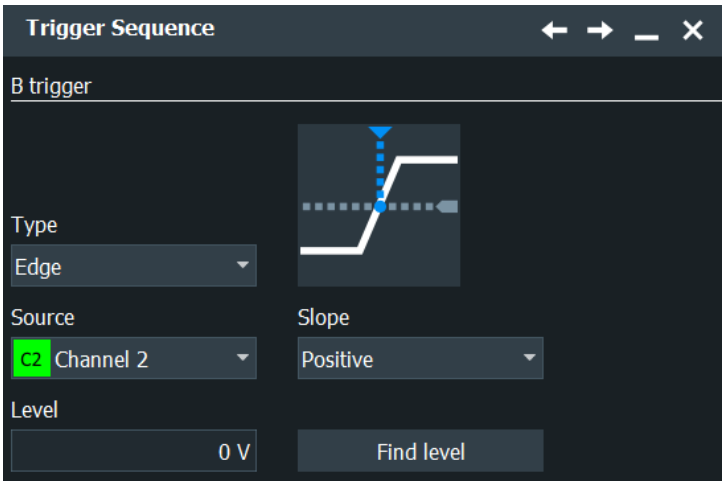
For the zone trigger, more trigger sequences are available, see Chapter 6.9, "Zone trigger", on page 232

6.8.1 Sequence setup

Access: "Menu">"Trigger" > "Setup" tab > "Trigger on" = "Sequence"



The B-trigger is the second condition of the trigger sequence. You can configure a delay between the A- and B-trigger, and define a number of fulfilled B-trigger conditions to be ignored. The B-trigger condition is configured in the same way as the A-trigger. The instrument disables settings that do not fit the previous settings in the sequence.



The reset condition R can be a timeout or a trigger condition, or a combination of both.

Sequence

Selects the type of the sequence.

"A → B → R" Triggers if all conditions of A- and B-events, as well as additional delay and reset timeout or R-event (reset) conditions are fulfilled.

"A → Zone"	Triggers if the conditions of the A-event and the zone trigger are fulfilled. A holdoff condition also can be set.
"A → B → R → Zone"	Triggers if all conditions of A- and B-events, reset timeout or R-event (reset) conditions are fulfilled for the defined zone.
"A OR B"	Triggers if the conditions of A-event or the conditions of the B-event are fulfilled.
"A OR B → Zone"	Triggers if the conditions of A-event or the conditions of the B-event are fulfilled for the defined zone. A holdoff condition also can be set.

Remote command:
`TRIGger<m>:SEQuence:TYPE` on page 1427

After 1st event wait
Sets the time that the instrument waits after an A-trigger until it recognizes B-triggers.

Remote command:
`TRIGger<m>:SEQuence:DELay` on page 1428

2nd event count
Sets the number of B-trigger conditions to be fulfilled after an A-trigger. The last B-trigger causes the trigger event.

The waiting time for B-triggers can be restricted with a reset condition: timeout or reset event.

Remote command:
`TRIGger<m>:SEQuence:COUNt` on page 1428

Reset on event/Reset event
If enabled, the trigger sequence is restarted by the R-trigger condition if the specified number of B-triggers does not occur before the trigger conditions are fulfilled.

The R-trigger condition is configured in the same way as the A-trigger. The instrument disables settings that do not fit the previous settings in the sequence.

Remote command:
`TRIGger<m>:SEQuence:RESet:EVENt` on page 1428

Reset on timeout/Reset timeout
If timeout is enabled, the instrument waits for the "Timeout" time for the specified number of B-triggers. If no trigger occurs during that time, the sequence is restarted with the A-trigger.

Remote command:
`TRIGger<m>:SEQuence:RESet:TIMeout[:ENABle]` on page 1429
`TRIGger<m>:SEQuence:RESet:TIMeout:TIME` on page 1429

Couple sequence thresholds
Sets the trigger levels to the values of the latest configured trigger condition. Each channel has its own trigger level.

Example:

If you have configured the A-trigger at last, and the trigger level for C1 is 70 mV, enabling the coupling sets the trigger levels for C1 in the B- and R trigger also to 70 mV. If the B-trigger and/or R-trigger use another source as the A-trigger, the level remains unchanged.

Remote command:
`TRIGger<m>:ECOupling` on page 1428

6.9 Zone trigger

The zone trigger triggers on the intersection or non-intersection of the signal and one or more zones or masks. The zone can be applied to any active input signal, math waveform including FFT, and XY-waveform.

You can use the zone trigger, for example, to solve the following tasks:

- Trigger if a peak in the spectrum occurs: define a zone in the FFT diagram to filter amplitude peaks. In the same way, you can filter harmonics.
- Separate rising and falling edges: define a zone around the base or top of the data signal.
- Separate read/write cycles: define a zone in the eye diagram.
- Identify a tube violation of signals with an infrequent non-monotonic edge.
- Filter events in the history, after acquisition has finished

To document the trigger events, use the actions on trigger. For example, create a report when the instrument triggers, or save the waveform.

Zone trigger is not available for serial protocol triggers.

Trigger sequence with zone trigger

You can combine the zone trigger with common trigger conditions and use one of the following trigger sequences:

- "A → Zone"
- "A → B → R → Zone"
- "(A OR B) → ZoneTrigger"

In a sequence, the zone trigger acts like a filter on previous acquisitions. Previous conditions and then the zone condition must be fulfilled. For example, in an "A → Zone" sequence, the trigger marker is set to the A event, and a corresponding trigger-out signal is sent when the zone condition is fulfilled.

6.9.1 About trigger zones

A zone is a mask without result box. You can adjust the shape of the zone in the same way as mask segments, graphically by dragging the corner points, or numerically in the

"Masks" > "Mask Definition" dialog box. You can also use existing masks in the zone trigger. The zone trigger and usual mask tests run in parallel.

If you switch off a mask test that is used by the zone trigger, the mask is removed from the zone trigger expression automatically.

All zones and masks that are included in the zone trigger expression are indicated with yellow color.

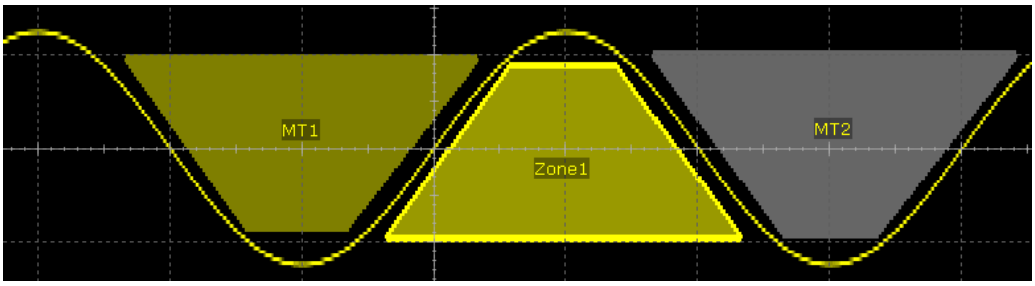


Figure 6-4: Indication of trigger zones

MT1 = Mask is included in zone trigger. Trigger if signal hits the mask (must intersect).
MT2 = Mask is not included in zone trigger.
Zone1 = Zone is included in zone trigger. Trigger if signal does not hit the zone (must not intersect).

If average or envelope acquisition is enabled, only triggered waveforms are used for envelope and average calculation on channel and math waveforms.

6.9.2 Creating trigger zones

You can define the trigger zones on the display, or use existing mask definitions as trigger zones.

Define trigger zones

1. Tap the "Zone trigger" icon on the toolbar.

2. In the signal bar, select if the signal must violate the zone to cause a trigger (Must Intersect), or vice versa (Must Not Intersect).
3. Tap the corner points of the zone on the screen.
4. If all points are set, tap "Finish zone" in the signal bar.

   "Zone1" is shown in the expression field.

5. If you want to define another zone, select the logic combination of the zones AND (both zones) or OR (either zone).
6. To create the second zone, repeat steps 3 and 4.
7. To define more zones, repeat steps 5 and 6.

Use existing masks for zone trigger

Prerequisites: At least one mask is created and visible on the display.

1. Tap the "Zone trigger" icon on the toolbar.

2. In the signal bar, select if the signal must violate the zone to cause a trigger (Must Intersect), or vice versa (Must Not Intersect).
3. Tap the mask on the display.
4. Tap "Use selected mask" in the signal bar.

   "MT1" is shown in the expression field.

5. If you want to add another mask test or zone, select the logic combination of the zones AND (both zones) or OR (either zone).
6. To add another mask, tap the mask on the display.
   Tap "Use selected mask" in the signal bar.
7. To add another zone, tap the corner points of the zone on the screen.
   If all points are set, tap "Finish zone" in the signal bar.

If a zone trigger is already defined, you can also add a new mask to the zone trigger when creating the mask.

Intersect
Defines if the signal must intersect the zone to allow the instrument to trigger, or if it must not intersect the zone.

Combine
Sets the logic combination of two zones.

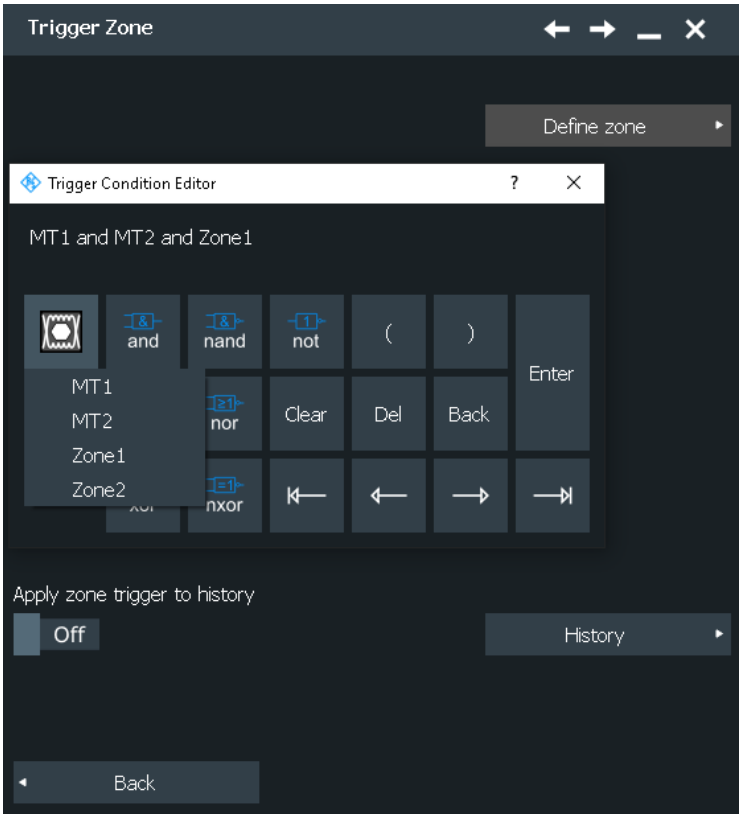
Use selected mask
Includes the selected zone or mask in the zone expression.

Clear Expr
Deletes the zone trigger expression. The zones are not deleted, they remain as usual masks, and the result boxes of the mask test appear.

Delete all
Deletes the zone trigger expression and the zones.

6.9.3 Advanced setup and analysis with zone trigger

Access: overlay menu for zone trigger > "Advanced Setup", or "Trigger" menu > "Setup" > select sequence with zone trigger > "Setup Z trigger" dialog



Logical trigger condition
If all required mask tests and zones are defined, you can type the logical expression directly, or use the trigger condition editor. All logical combinations are available in the editor. To express the "Must Not Intersect" condition, use the logic NOT.

Remote command:
`TRIGger<m>:ZONE:EXPRession[:DEFine]` on page 1433

Apply zone trigger to history
Applies the zone trigger condition to the acquisitions in the history memory.

The history saves all acquisitions that fulfill the trigger condition that is set during acquisition (zone trigger condition, or another trigger condition). If "Apply zone trigger to history" is disabled, the history "Play" reads and displays all saved acquisitions from the memory.

If the setting is enabled, the zone trigger is applied to the history replay. "Play" displays only acquisitions that fulfill the zone trigger condition.

Using "Apply zone trigger to history", you can:
- Acquire waveforms with high speed and filter them afterwards in the history.
- Change the zone trigger condition after acquisition.

Remote command:
`SWTRigger:HISTory` on page 1433

6.10 External trigger input

Except for using analog or digital input channels as trigger source, you can also use external signals as trigger source. The external signal is connected to the external trigger input, which has a BNC connector. The only trigger type to trigger on external signals is the edge trigger.

1. Connect the external trigger signal to the external trigger input.
2. Set up the trigger:
   a) Select the "Trigger" > "Setup" tab.
   b) Select the source: "Extern"
   c) Adjust the trigger settings.
   See Chapter 6.10.1, "External trigger setup", on page 236.

If the trigger source is a channel input, the trigger system uses the digitized signal. The trigger system of the instrument is a separate system, thus the signal processing by enhancement, decimation and arithmetic has no impact on the trigger signal. Most of the R&S RTO6 trigger types use the digitized trigger signal.

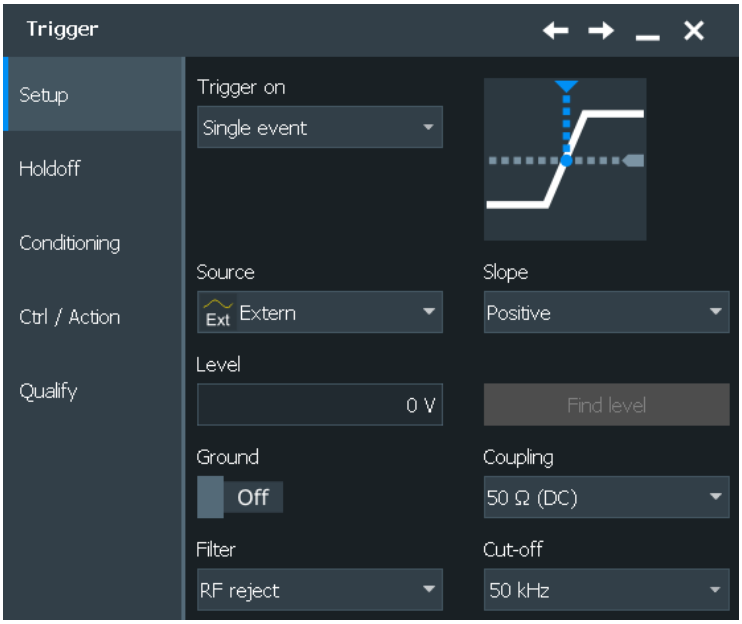
If the trigger source is the external trigger input, the trigger comparator uses the analog input signal. For the external trigger signal, only the edge trigger of the A-trigger is available. Trigger sequence is not supported.

Qualification of the external trigger signal is not available.

6.10.1 External trigger setup

Access: "Menu" > "Trigger" > "Setup" tab > "Source = Extern"

External trigger signals, which are connected to the external trigger input, can be triggered with an edge trigger. The "Find level" function is not available for external trigger signals.



Ground
If the selected trigger source is the external trigger input, you can connect the trigger input to the ground.

Remote command:
`TRIGger<m>:ANEDge:GND` on page 1436

Coupling
You can set the coupling in the trigger configuration.

"50 Ω (DC)"	Direct connection with 50 Ω termination, passes both DC and AC components of the trigger signal.
"1 MΩ (DC)"	Direct connection with 1 MΩ termination, passes both DC and AC components of the trigger signal.
"1 MΩ (AC)"	Connection through capacitor, removes unwanted DC and very low-frequency components.

Remote command:
`TRIGger<m>:ANEDge:COUPling` on page 1434

Filter
If the selected trigger source is "Extern" (external trigger input), you can directly select a filter to reject high or low frequencies.

For all other trigger sources, you can add a digital filter using the Digital Filter Setup.

"Off"	The trigger signal is not filtered.
"LF reject"	Frequencies below the "Cut-off" frequency are rejected, higher frequencies pass the filter. You can adjust the "Cut-off" frequency, the default is 50 kHz.

"RF reject" Frequencies higher than the "Cut-off" frequency are rejected, lower frequencies pass the filter. You can adjust the "Cut-off" frequency, the default is 50 kHz.

Remote command:
`TRIGger<m>:ANEDge:FILTer` on page 1434
`TRIGger<m>:ANEDge:CUToff:HIGHpass` on page 1435
`TRIGger<m>:ANEDge:CUToff:LOWPass` on page 1435

Slope
Sets the edge type for the trigger condition.

"Positive" Selects the rising edge, that is a positive voltage change.

"Negative" Selects the falling edge, that is a negative voltage change.

"Both" Selects the rising as well as the falling edge. This option is not available if the trigger source is the external trigger input.

Remote command:
`TRIGger<m>:EDGE:SLOPe` on page 1392
`TRIGger<m>:ANEDge:SLOPe` on page 1436
`TRIGger<m>:SLEW:SLOPe` on page 1403

Trigger level
Sets the voltage level for the trigger condition. You can also drag the trigger level marker on the display (TA or TB on the right edge of the display). The range of the trigger level is limited in a way so that always a hysteresis for stable trigger conditions is available.

Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1391

7 Waveform analysis

This chapter describes general methods to check and analyze waveforms. These are:

- Zoom....239
- Reference waveforms....249
- Mathematics....256
- History....274
- XY-diagram....280

7.1 Zoom

The zoom functions allow you to magnify a specific section of the diagram in order to view more details. You can define several zoom areas for the same diagram and even couple them, or you use the hardware zoom.

7.1.1 Methods of zooming

The R&S RTO6 provides various ways of zooming: You define the section of a diagram that you want to magnify, and the zoomed view is shown in a separate zoom diagram. Additionally, you can magnify the diagram directly: The hardware zoom changes the horizontal and vertical scales of the diagram so that you see the selected section.

There are different ways to initiate and configure the zoom function:

- **Fingertip zoom**: magnifies the waveforms around your fingertip. When you drag your finger, the magnifier moves, too. You can convert the fingertip zoom into a standard zoom diagram.
- **Graphical method**: you draw, move and adjust the zoom area on the touchscreen – a very quick and simple method for standard zoom and hardware zoom.
- **Numeric method**: you enter x- and y-values in a dialog box or adjust them using navigation controls. These are precise ways which can be used to optimize a graphically defined zoom.
  With the numeric method there are two ways of defining the zoom area:
  – Specifying **start and stop values** for the x- and y-axes; the acquired data within those values is zoomed.

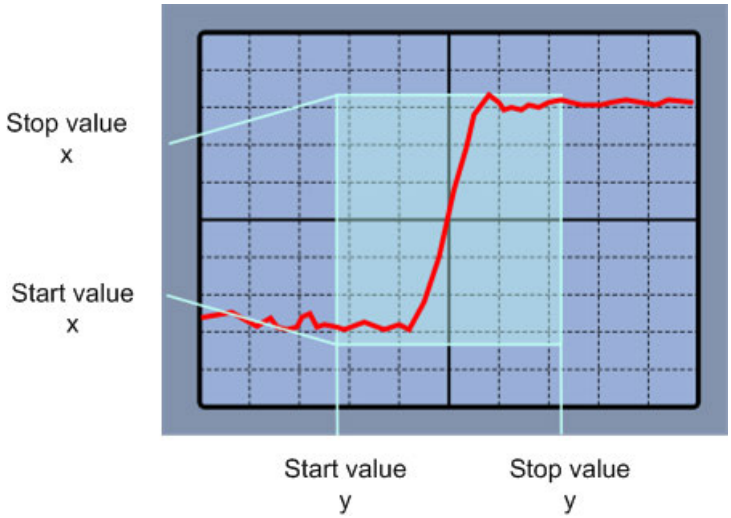


Figure 7-1: Numeric zoom using start and stop values

– Specifying the x- and y-**position** of the centerpoint of the area plus a **range** for the x- and y-axes; the area defined by that centerpoint and the ranges is zoomed.

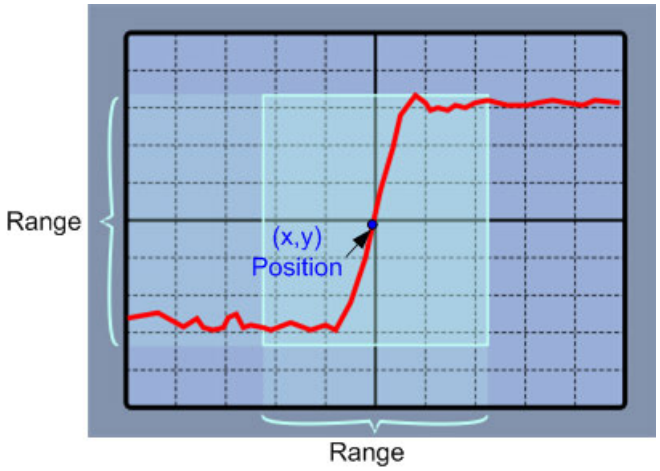


Figure 7-2: Numeric zoom using position and range

- **Coupled zoom** creates a copy of the selected zoom area. Coupled zoom areas always have the same size (size coupling). They can be positioned separately or together (position coupling).

Zoom areas can be used for gating, for example, to define a measurement gate. You can set the gate exactly to the limits of the zoom.

Evaluation gates - available histogram areas, masks, and measurement gates - can be displayed in zoom diagrams to simplify the graphical gate adjustment on the touch-screen.

See: Chapter 7.1.3, "Zooming for details", on page 245.

7.1.2 Zoom settings

The zoom area, i.e. the section to be enlarged, can be defined using two different methods:

- Using the zoom functions on the toolbar and draw the zoom area on the touch-screen
- Specifying numeric values:
  - start and stop values for the x- and y-axes
  - x and y position of one point in the diagram plus a range for the x- and y-axes

  See also: Chapter 7.1.1, "Methods of zooming", on page 239.

- Zoom functions on the toolbar.....241
- Zoom settings.....241

7.1.2.1 Zoom functions on the toolbar

The zoom icon on the toolbar shows the last selected zoom type. A short tap on the icon activates the selected zoom. If you touch the icon and drag your finger down, a menu opens where you can select another zoom type.

Standard zoom
Displays a magnified section of the diagram in an additional zoom diagram. It is a display zoom, instrument settings are not changed.

Touch and hold the zoom area to open the "Zoom" dialog box.

Remote command:
`LAYout:ZOOM:ADD` on page 1438

Hardware zoom
Changes the instrument settings - horizontal and vertical scales as well as trigger level and offset - to display a part of the diagram in greater detail.

Coupled zoom
Creates a coupled zoom area and its related zoom diagram. If you change the size of one zoom area, the size of all coupled zoom areas is changed as well.

Remote command:
`LAYout:ZOOM:ADDCoupled` on page 1438

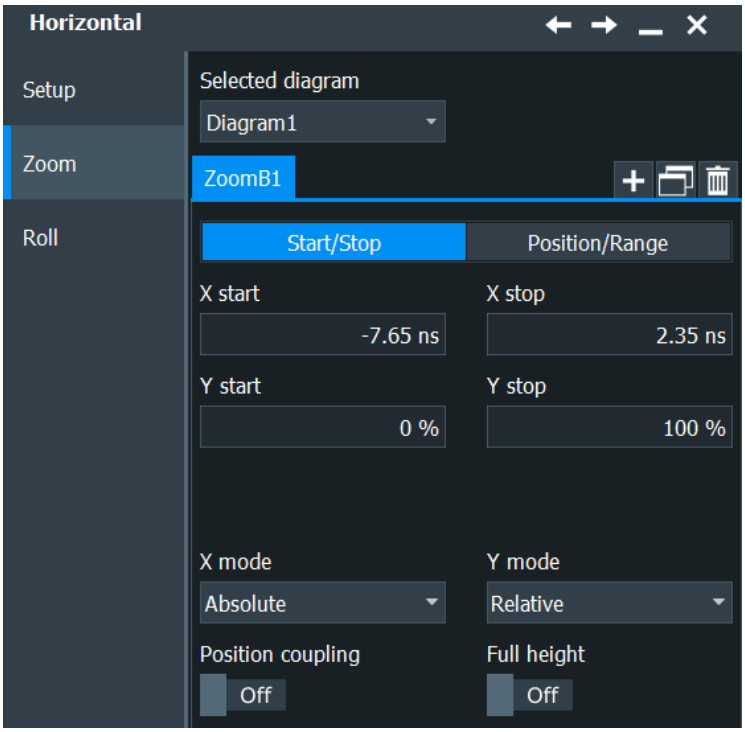
Fingertip zoom
Magnifies the waveforms around your fingertip.

Tap the icon and put your finger on the waveform. The touched part of the waveform is displayed in a magnifier. Drag your finger on the screen to move the magnifier. You can change the zoom factor using the [Multiuse] knob.

7.1.2.2 Zoom settings

Access: "Menu" > "Horizontal" > "Zoom" tab.

The "Zoom" tab allows you to specify start and stop values for the x- and y-axes. The acquired data within these ranges is zoomed.

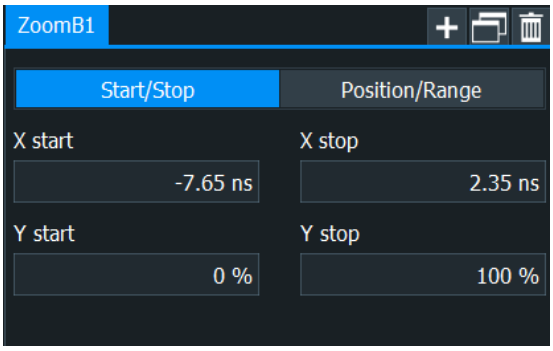


Additionally, you can enable a "Zoom Overlay" in the "Menu" > "Settings" > "Appearance" > "Diagram", see Chapter 4.3.4, "Diagram appearance settings", on page 79.

Selected diagram
Indicates which of the waveform diagrams is selected for zooming.

Mode
Selects how the window for the zoom diagram is defined. You can select between defining "Start/Stop" values or "Position/Range".

Start/Stop
The "Start/Stop" tab allows you to specify start and stop values for the x- and y-axes. The acquired data within these ranges is zoomed.



X start ← Start/Stop
Defines the lower limit of the zoom area on the x-axis.

Remote command:
`LAYout:ZOOM:HORZ:ABSolute:STARt` on page 1440
`LAYout:ZOOM:HORZ:RELative:STARt` on page 1442

X stop ← Start/Stop
Defines the upper limit of the zoom area on the x-axis.

Remote command:
`LAYout:ZOOM:HORZ:ABSolute:STOP` on page 1441
`LAYout:ZOOM:HORZ:RELative:STOP` on page 1442

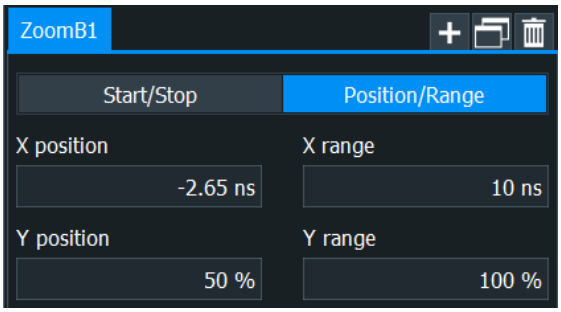
Y start ← Start/Stop
Defines the lower limit of the zoom area on the y-axis.

Remote command:
`LAYout:ZOOM:VERTical:RELative:STARt` on page 1445
`LAYout:ZOOM:VERTical:ABSolute:STARt` on page 1444

Y stop ← Start/Stop
Defines the upper limit of the zoom area on the y-axis.

Remote command:
`LAYout:ZOOM:VERTical:RELative:STOP` on page 1445
`LAYout:ZOOM:VERTical:ABSolute:STOP` on page 1444

Position/Range
In the "Position/Range" tab, you specify the x and y position of center point of the zoom area plus a range for the x- and y-axes; the area defined by that point and the ranges is zoomed.



X position ← Position/Range
Defines the x-value of the centerpoint of the zoom area.

Remote command:
`LAYout:ZOOM:HORZ:ABSolute:POSition` on page 1439
`LAYout:ZOOM:HORZ:RELative:POSition` on page 1441
`SEARch:RESDiagram:HORZ:ABSolute:POSition` on page 1634
`SEARch:RESDiagram:HORZ:RELative:POSition` on page 1635

X range ← Position/Range
Defines the width of the zoom area.

Remote command:
`LAYout:ZOOM:HORZ:ABSolute:SPAN` on page 1440
`LAYout:ZOOM:HORZ:RELative:SPAN` on page 1441
`SEARch:RESDiagram:HORZ:ABSolute:SPAN` on page 1635
`SEARch:RESDiagram:HORZ:RELative:SPAN` on page 1636

Y position ← Position/Range
Defines the y-value of the centerpoint of the zoom area.

Remote command:
`LAYout:ZOOM:VERTical:ABSolute:POSition` on page 1443
`LAYout:ZOOM:VERTical:RELative:POSition` on page 1444
`SEARch:RESDiagram:VERT:ABSolute:POSition` on page 1636
`SEARch:RESDiagram:VERT:RELative:POSition` on page 1637

Y range ← Position/Range
Defines the height of the zoom area.

Remote command:
`LAYout:ZOOM:VERTical:RELative:SPAN` on page 1445
`LAYout:ZOOM:VERTical:ABSolute:SPAN` on page 1443
`SEARch:RESDiagram:VERT:ABSolute:SPAN` on page 1636
`SEARch:RESDiagram:VERT:RELative:SPAN` on page 1637

X mode
Defines whether absolute or relative values are used to specify the x-axis values.

Remote command:
`LAYout:ZOOM:HORZ:MODE` on page 1439
`SEARch:RESDiagram:HORZ:MODE` on page 1635

Y mode
Defines whether absolute or relative values are used to specify the y-axis values.

Remote command:
`LAYout:ZOOM:VERTical:MODE` on page 1442
`SEARch:RESDiagram:VERT:MODE` on page 1637

Position coupling
Enables or disables the position coupling of coupled zooms. If position coupling is enabled and you move one zoom area, the other coupled zoom areas are moved, too, and keep their distance.

Remote command:
`LAYout:ZOOM:POSCoupling` on page 1439

Full height
Uses the full diagram height for the zoom area. Only horizontal zoom settings can be changed.

7.1.3 Zooming for details

The usage of the various zoom methods is described in the following procedures:

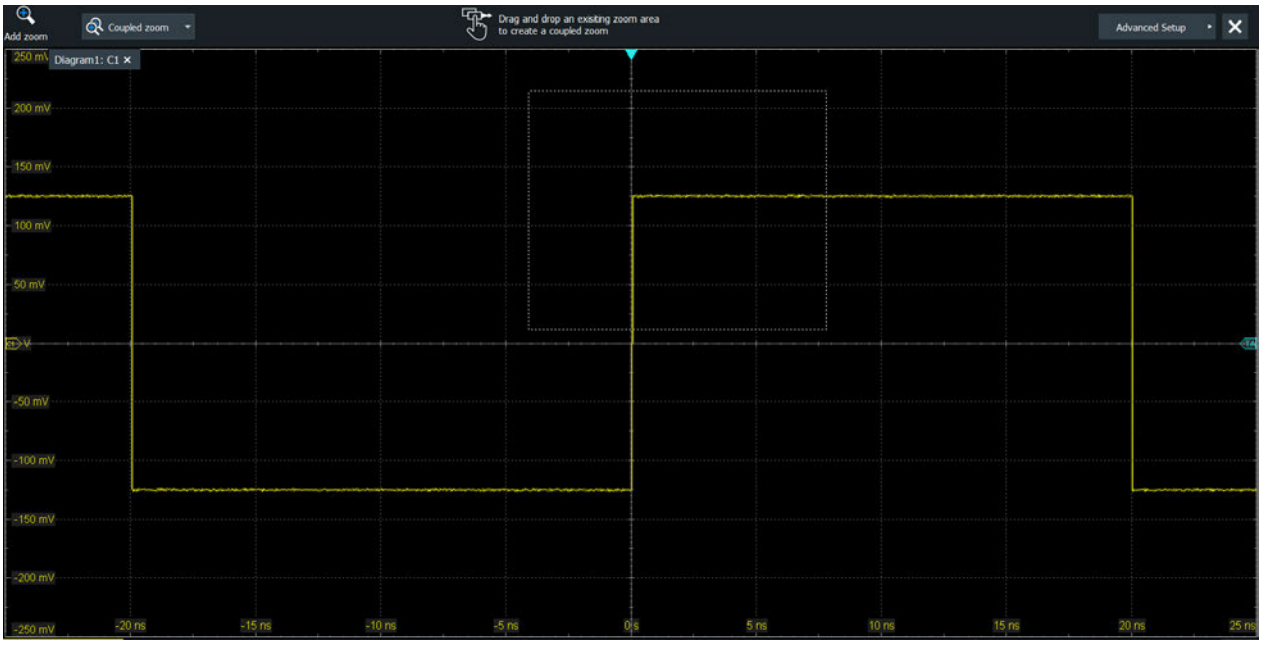
- To define the zoom area graphically on the touchscreen
- To define the zoom area numerically using start/stop values
- To define the zoom area numerically using position and range values
- To define multiple zoom areas
- To define coupled zoom areas

To define the zoom area graphically on the touchscreen

For graphical zooming, you use your finger on the screen.

1. On the toolbar, tap the "Standard Zoom" icon.

2. Touch the position that you want to define as one corner of the zoom area. Then drag your finger to the opposite corner of the zoom area.
   While you drag your finger on the touchscreen, a dotted rectangle indicates the current zoom area. When the rectangle covers the required zoom area, remove your finger.



The indicated area is magnified in a new zoom diagram. The original diagram is displayed with the zoom area indicated as a rectangle.

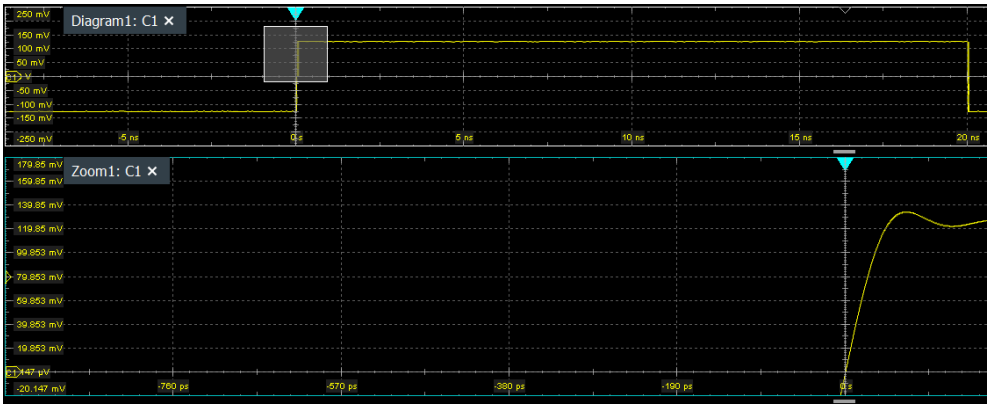


Figure 7-3: Zoom diagram and overview diagram

3. If the position of the zoom area is not correct, drag the rectangle in the overview to the correct position.
4. If the size of the zoom area is not yet ideal, tap the rectangle in the overview diagram.
   Now, 4 red lines indicate the edges of the zoom area. A dashed red line indicates the selected edge, which you can adjust.
5. Touch the edge that you want to move, and drag it to the required position.

To optimize the zoom definition of an active zoom diagram, double-tap the zoom diagram. The "Zoom" dialog box for numeric definition is opened.

To create a new zoom using the Zoom dialog box

1. There are two ways to create a new zoom:
   - If you want to create a new, unconfigured zoom, tap the "Add" icon.

   - If you want to create zoom based on an existing one, tap the "Copy" icon.

2. Enter a name for the zoom using the on-screen keyboard.

To define the zoom area numerically using start/stop values

1. On the "Menu" menu, tap "Horizontal".
2. In the "Zoom" tab, select "Start/Stop".
3. Select "X mode" to define "Absolute" or "Relative" x-axis values. Relative values cause the zoom area to adapt to the input values dynamically.
4. Define the "X start" and "X stop" values that define the lower and upper borders of the zoom area on the x-axis (see Figure 7-1).

5. Select the "Y mode" to define "Absolute" or "Relative" y-axis values.
6. Define the "Y start" and "Y stop" values that define the lower and upper borders (respectively) of the zoom area on the y-axis.

   When you close the dialog box, the specified area is magnified in a new zoom diagram. The original diagram is displayed with the zoom area indicated as a rectangle (see Figure 7-3).

To define the zoom area numerically using position and range values

1. On the "Menu" menu, tap "Horizontal".
2. In the "Zoom" tab, select "Position/Range".
3. Select "X mode" to define "Absolute" or "Relative" x-axis values. Relative values cause the zoom area to adapt to the input values dynamically.
4. Under "X position", define the x-value of the center point of the zoom area (see Figure 7-2).
5. Under "X range", define the width of the zoom area.
6. Select the "Y mode" to define "Absolute" or "Relative" y-axis values.
7. Under "Y position", define the y-value of the center point of the zoom area.
8. Under "Y range", define the height of the zoom area.

   When you close the dialog box, the specified area is magnified in a new zoom diagram. The original diagram is displayed with the zoom area indicated as a rectangle.

To define multiple zoom areas

You can define more than one zoom area for the same diagram, for example to compare several peaks in a measurement. These zoom areas can be displayed in separate zoom diagrams, or together in one zoom diagram.

To define multiple zoom areas graphically, simply repeat the steps described in To define the zoom area graphically on the touchscreen - for each area. Numerically, proceed as follows:

1. On the "Menu" menu, tap "Horizontal".
2. Select the "Zoom" tab.
3. Select the required tab according to the method you want to use to define the zoom area.
4. To copy the current zoom area definition, tap the "Copy" icon.
   Alternatively, tap the "Add" icon to add a new zoom area.
5. Enter a name for the new zoom diagram using the displayed on-screen keyboard.
6. Define the zoom area as described for the first zoom.

An additional zoom diagram is displayed for the new zoom area, and another rectangle in the original diagram indicates the new zoom area. Each rectangle in the overview has the same color as the corresponding zoom diagram frame.

7. Alternatively, you can overlay the zoom areas: The zooms are shown in the same zoom diagram, as if the zoom areas are overlaid:
   In the "Settings" > "Appearance" dialog > "Diagram " tab , enable "Zoom > Overlay".

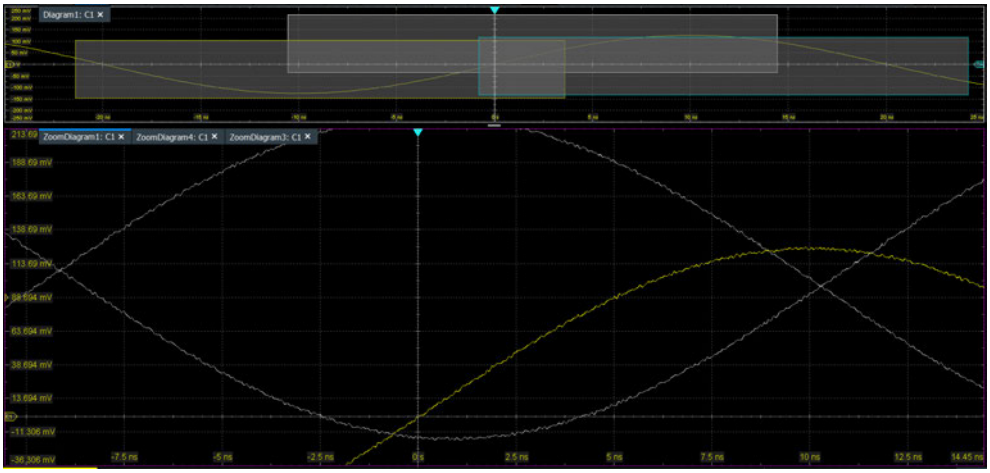


Figure 7-4: Multiple zoom diagrams. Left: separate zoom diagrams. Right: overlaid zoom

To define coupled zoom areas

You can define multiple zoom areas for one diagram that are coupled. If you change the size of one zoom area, the size of all coupled zoom areas is changed as well. Furthermore, you can couple also the position in order to move all coupled zooms at once. Coupling is useful, for example, if you want to compare recurring peaks in a signal.

1. On the toolbar, tap the "Coupled Zoom" icon.

2. In the diagram overview, tap an existing zoom area.

   The selected zoom area is duplicated.

3. Drag the duplicate zoom area to the required position.
4. To create further coupled zooms, repeat the steps above.

   Now, if you change the zoom area size of any of the coupled zoom areas in the "Zoom" dialog box, the settings are changed for all coupled zoom areas.

5. In the "Zoom" dialog box, select the diagram that contains the coupled zooms.
6. Select a zoom tab.
7. Enable "Position coupling".

If you move one of the coupled zoom areas in the diagram, all other coupled zooms are moved as well, and their distance is kept unchanged.

To use the hardware zoom

In contrast to the normal zoom, the hardware zoom changes the instrument settings - horizontal and vertical scales, and also the trigger level and offset. Thus, the selected area is displayed in the diagram instead of the original waveform. No additional zoom diagram is opened.

1. On the toolbar, tap the "Hardware Zoom" icon.

2. Drag your finger on the touch screen to mark the zoom area.
   A dotted rectangle indicates the current zoom area. When the rectangle covers the required zoom area, remove your finger. The diagram changes and shows the magnified area.

   Tip: To return to the previous display, use the "Undo" icon.

   Note: You can combine hardware zoom and normal zoom - first use the hardware zoom, then the zoom into the display. The reverse approach is also possible: Create a zoom diagram, and then apply the hardware zoom to the waveform diagram. Both the waveform and the zoom diagrams are changed.

7.2 Reference waveforms

You can configure up to four reference waveforms to display stored waveforms. Any active signal or mathematical waveform can be stored as a reference waveform. It can then be loaded again later to restore the waveform on the screen.

7.2.1 Working with reference waveforms

Reference waveforms can be displayed in addition to the signal waveforms, saved to file, and loaded back for further analysis. Reference waveforms can be loaded only from BIN files.

Note: Saving and loading reference waveforms, and preset with active reference waveform delete the undo stack. After these actions, undo is not possible.

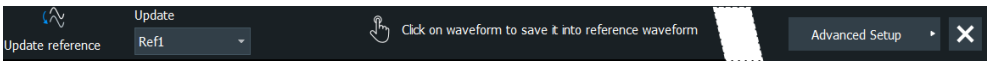
To update a reference waveform using the toolbar icon

If you often need to update a reference waveform, you can use the "Save Reference" toolbar icon.

1. Add the "Save Reference" icon to the toolbar, see Chapter 3.3.6.2, "Configuring the toolbar", on page 53 .

2. Tap the icon.

   The overlay menu opens.



3. Set "Update" to the reference waveform to be used.
4. Tap the waveform to be used as reference waveform.

To display a reference waveform

1. In the "Menu" > "Apps" > "Analysis" tab, select "Reference Curve".
   Alternatively, press the [Ref] key.
2. Select the tab for the reference waveform you want to display ("R1"-"R4").
3. Load a stored reference waveform as described in "To load a reference waveform" on page 251, or select a source to be displayed as a reference:
   a) In the "Setup" tab, select "Signal".
   b) Select the "Source" from the selection list. The source can be any active signal, math, or other reference waveform.
   c) Tap the "Update" button to update the current reference waveform with the source data.
4. Tap the "Show" button.

   The reference waveform is displayed on the screen.
5. A reference waveform can have its own scaling settings or it can be scaled according to the source settings. By default, the scaling of the reference waveform is coupled to the source settings. Additionally, it can be stretched or compressed in vertical and horizontal direction.
   If necessary, change the settings on the "Vertical" and "Rescale" tabs of the "Reference Waveform" dialog boxes.
   To restore the original settings, tap the "Restore settings" in the "Vertical" tab.
   For a description of the scaling settings, see Chapter 7.2.2.4, "Rescale", on page 255

To save a reference waveform

1. In the "Save/Recall" dialog, select "Save > Waveform".

   Tip: You can also save a waveform as a reference waveform in the "File" dialog box, see Chapter 12.2.7, "Saving and loading waveform data", on page 473. Here, you can also save multiple waveforms in one file.
2. Select the tab for the reference waveform you want to store ("Ref1"-"Ref4").
3. Display and configure the reference waveform as described in "To display a reference waveform" on page 250.

4. Select the file format.

   Note: Reference waveforms can be loaded only from BIN files. XML and CSV formats are meant for further processing in other applications.

5. To save the waveform to the currently selected file, tap "Save". By default, the prefix for reference waveform files is "RefCurve".
   To save the waveform to another file, select "Save As".
   Enter a file name and select the directory. The file type is already defined according to the selection in the previous step. In order to load the reference waveform on the instrument again later, use the file type BIN.

   The source settings of the reference waveform and the current scaling settings are stored to the specified file.

To load a reference waveform

Note: Reference waveforms can be loaded only from BIN files.

1. Press the [Ref] key.
2. Select the tab for the reference waveform you want to load ("Ref1" - "Ref4").
3. In the "Setup" tab, select "File".
4. To load the waveform from the specified file, tap "Open".
   To load the waveform from a different file, tap "Open". Select the file from the file selection dialog box. Only BIN files are displayed in the file list.

   The selected waveform is loaded as the specified reference waveform.
   If multiple waveforms are saved in the file, you are asked to assign each waveform to a reference waveform. All waveforms are loaded together.

To view a reference waveform

► You can view a reference waveform, using the "Saveset with preview" function.
For details, see: Chapter 12.1.4, "Saveset with preview function", on page 449.

7.2.2 Settings for reference waveforms

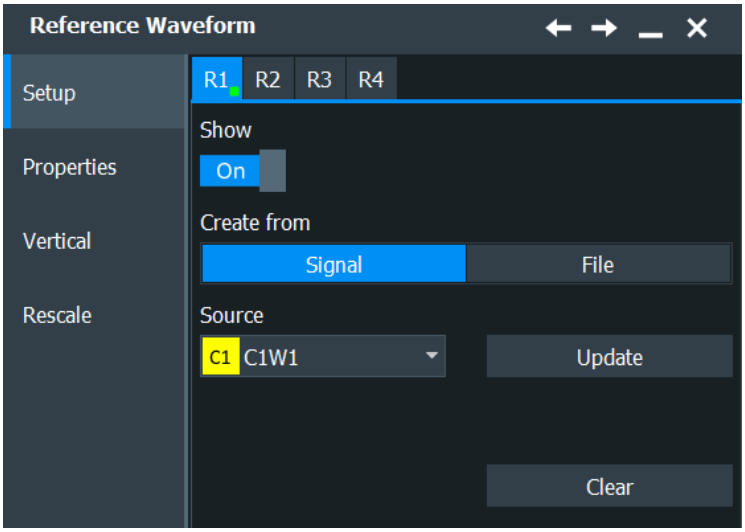
To compare waveforms and analyze differences between waveforms, you can use up to four reference waveforms R1 to R4. Each reference waveform has its own memory on the instrument. You can also save an unlimited number of reference waveforms and load them for further use.

The display of a reference waveform is independent from the display of the source waveform; you can move, stretch and compress the curve vertically and horizontally.

7.2.2.1 Reference waveform setup

Access: [Ref] key > "Setup" tab

In the "Setup" tab, you select the reference waveform and its source. The source can be an active waveform - trace of an input channel, math waveform or another reference waveform - or a stored waveform.



R1/2/3/4 252
Show 252
Signal 252
└ Source 252
└ Update 252
File 253
Clear 253

R1/2/3/4
Each tab contains the settings for one of the four available reference waveforms.

Show
Displays the reference waveform in the diagram.

Remote command:
`REFCurve<m>:STATe` on page 1447

Signal
Selects a signal as the reference waveform.

Source ← Signal
Selects the source waveform from the active waveforms of input channels, math signals and other reference waveforms.

Remote command:
`REFCurve<m>:SOURce` on page 1446

Update ← Signal
Copies the selected source waveform with all its settings to the memory of the reference waveform. If the acquisition is running, the reference waveform is a snapshot.

Remote command:
`REFCurve<m>:UPDate` on page 1448

File
Opens a file selection dialog box and loads the selected reference waveform file.

Double-tap the filename to open the file selection dialog box, see also Chapter 12.5, "File selection dialog", on page 482.

Note: Note that reference waveforms can be loaded from `.bin` files only. xml and csv formats are meant for further processing in other applications.

Remote command:
`REFCurve<m>:OPEN` on page 1447
`REFCurve<m>:SAVE` on page 1448
`REFCurve<m>:DELete` on page 1448

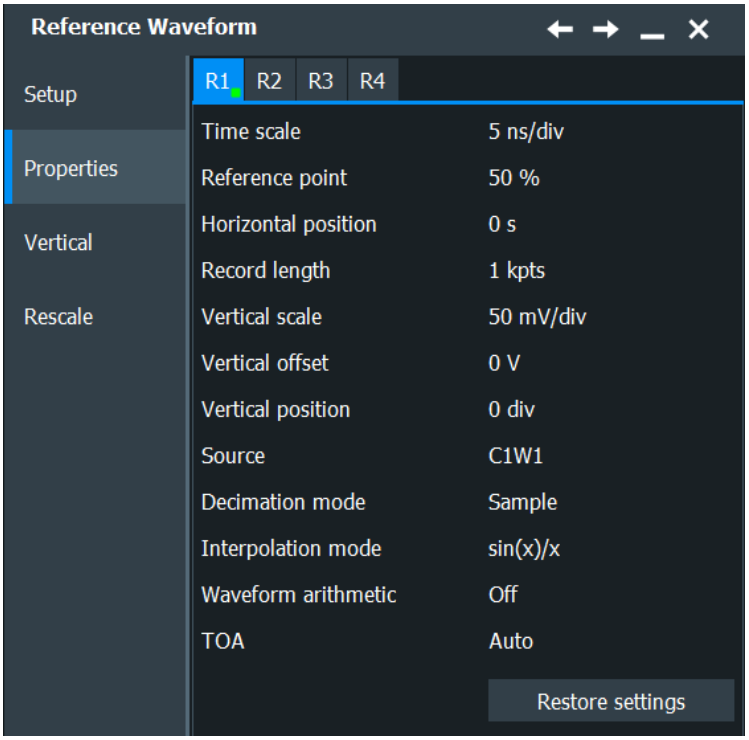
Clear
The selected reference waveform disappears, its memory is deleted.

Remote command:
`REFCurve<m>:CLEar` on page 1448

7.2.2.2 Properties

Access: [Ref] key > "Properties" tab

A reference waveform can be scaled, stretched and positioned in the diagram. The "Original Attributes" tab shows the original settings of the reference waveform, which are stored together with the waveform data.

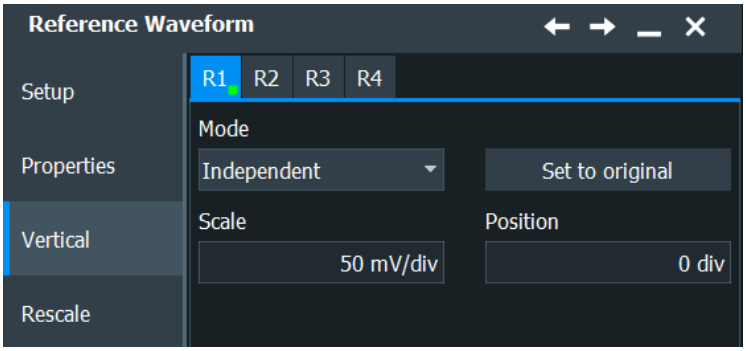


Restore Settings
Resets the time scale and the reference point to the original values of the reference waveform.

Remote command:
`REFCurve<m>:RESTore` on page 1449

7.2.2.3 Vertical

Access: [Ref] key > "Vertical" tab



Mode
Selects the type of vertical settings:

"Coupled to source" Vertical position and scale of the source are used.

"Independent" Scaling and position can be set specific to the reference waveform.

Remote command:
`REFCurve<m>:VMODe` on page 1449

Set to original
Restores the original vertical settings of the reference waveform (vertical scale, position, and offset), if vertical scaling is set to "Independent".

Remote command:
`REFCurve<m>:TOORignal` on page 1450

Scale
Sets the vertical scale for the reference waveform, if vertical scaling is set to "Independent". You can also use the vertical [Scale] knob to adjust this value.

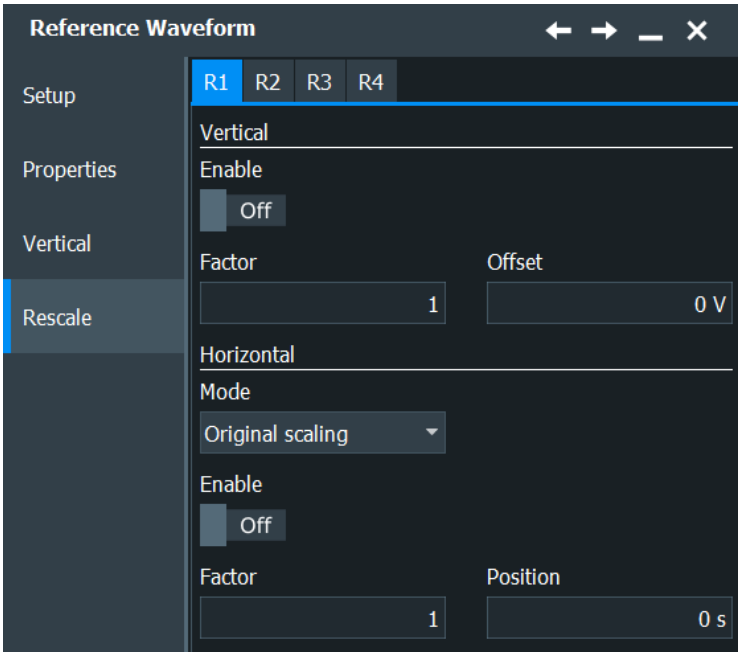
Remote command:
`REFCurve<m>:SCALe` on page 1449

Position
Moves the reference waveform up or down in the diagram, if vertical scaling is set to "Independent". If "Position" is assigned to the vertical [Position] knob, you can also use this knob to adjust the "Vertical position"

Remote command:
`REFCurve<m>:POSition` on page 1450

7.2.2.4 Rescale

Access: [Ref] key > "Rescale" tab



Vertical Rescaling
Stretching and offset change the display of the waveform independent of the vertical scale and position.

Enable ← Vertical Rescaling
If enabled, the vertical offset and factor are applied to the reference waveform.

Remote command:
`REFCurve<m>:RESCale:VERTical:STATe` on page 1450

Factor ← Vertical Rescaling
A factor greater than 1 stretches the waveform vertically, a factor lower than 1 compresses the curve.

Remote command:
`REFCurve<m>:RESCale:VERTical:FACTor` on page 1450

Offset ← Vertical Rescaling
Moves the reference waveform vertically. Enter a value with the unit of the waveform. Like vertical offset of a channel waveform, the offset of a reference waveform is subtracted from the measured value. Negative values shift the waveform up, positive values shift it down.

Note: As for all waveforms, a vertical offset of a reference waveform can be set using the vertical [Position] knob. This offset is independent from the reference scaling offset, which is described here. If both offsets are set, their values are added up.

Remote command:
`REFCurve<m>:RESCale:VERTical:OFFSet` on page 1451

Mode ← Vertical Rescaling
Selects the type of horizontal settings:

"Adjust to X Axis"	The current horizontal settings of the diagram are used.
"Original Scaling"	Horizontal scaling and reference point of the source waveform are used.

Remote command:
`REFCurve<m>:HMODe` on page 1451

Horizontal Rescaling ← Vertical Rescaling
Stretching and offset change the display of the waveform independent of the horizontal settings of the source waveform and of the horizontal diagram settings.

Enable ← Horizontal Rescaling ← Vertical Rescaling
If enabled, the horizontal offset and factor are applied to the reference waveform.

Remote command:
`REFCurve<m>:RESCale:HORizontal:STATe` on page 1452

Factor ← Horizontal Rescaling ← Vertical Rescaling
A factor greater than 1 stretches the waveform horizontally, a factor lower than 1 compresses the curve.

Remote command:
`REFCurve<m>:RESCale:HORizontal:FACTor` on page 1452

Position ← Horizontal Rescaling ← Vertical Rescaling
Moves the waveform horizontally. Enter a value with a time unit suitable for the time scale of the diagram. Positive values shift the waveform to the right, negative values shift it to the left.

Remote command:
`REFCurve<m>:RESCale:HORizontal:POSition` on page 1453

7.3 Mathematics

Math waveforms are calculated waveforms. You can define up to eight math waveforms and display them on the screen, and use it as source for further analysis.

Math waveforms are defined by mathematical expressions (formulas). You can enter mathematical expressions using different methods:

- "Operator" tab: you define a simple math function in a graphical editor by selecting the source waveforms and the operator.
- "Equation" tab: you define sophisticated math functions in a formula editor, as required to your needs.

The result of an FFT analysis is a specific math waveform. For information on FFT and spectrograms, see Chapter 9, "Spectrum analysis", on page 370.

The vertical scale of a math waveform is adapted automatically to the measurement results to ensure optimal display. Furthermore, you can scale each math waveform manually in vertical direction like a channel waveform.

As for channel waveforms, you can also change the arithmetic mode for the waveform to display the envelope or an average over several calculations.

You can store a math waveform as a reference waveform and restore it later, see "To save a reference waveform" on page 250.

- Displaying math waveforms....257
- Math setup - general settings....258
- Operator editor....259
- Equation....261
- Filters....269
- Math scale settings....271
- Math arithmetic....272

7.3.1 Displaying math waveforms

Math waveforms can be displayed in addition to the channel and other waveforms. They also can be used for analysis, e.g. measurements, even if the math waveform is not active.

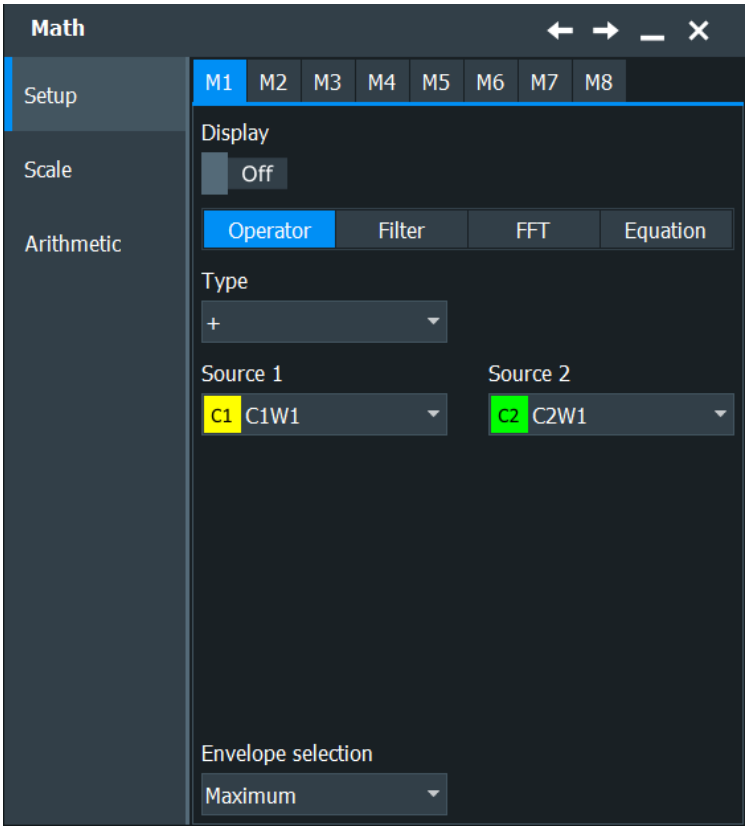
1. In the "Math" menu, select "Math Setup".
   Alternatively, press the [Math] key.
2. Define the math expression for calculation in one of the following ways:
   - Chapter 7.3.3.2, "Defining a formula in the Operator editor", on page 260
   - Chapter 7.3.4, "Equation", on page 261
   - Chapter 9.1.2, "Configuring spectrum waveforms", on page 374
3. In the "Math Setup" dialog box, in the "Setup" tab, tap the "Enable math signal".

   The math waveform is displayed on the screen.
4. To change the vertical scaling of the math waveform, tap the "Manual" icon.
5. Enter the "Vertical scale" factor (per division). If necessary, add a "Vertical offset". By default the instrument performs an automatic scaling.

   Tip: You can also use the vertical [Scale] rotary knob for scaling. In this case, the scale mode is set to "Manual" temporarily.
6. If you need the envelope or average of the math waveform over several calculations, change the arithmetic mode for the waveform as for channel waveforms. See also: "Arithmetic" on page 130.
7. Close the "Math Setup" dialog box.

7.3.2 Math setup - general settings

You can define up to eight different math waveforms. Each waveform is defined in a separate tab in the "Math" dialog box ("M1" to "M8").



The settings for input of mathematical formulas are described in separate chapters:

- Chapter 7.3.3.1, "Settings in the Operator editor", on page 259
- Chapter 7.3.5, "Filters", on page 269
- Chapter 9.1.3, "FFT configuration settings", on page 378
- Chapter 7.3.4, "Equation", on page 261

Display 258
Envelope selection 258

Display
If activated, a diagram for the defined math waveform is displayed on the touch screen.

Remote command:
`CALCulate:MATH<m>:STATe` on page 1456

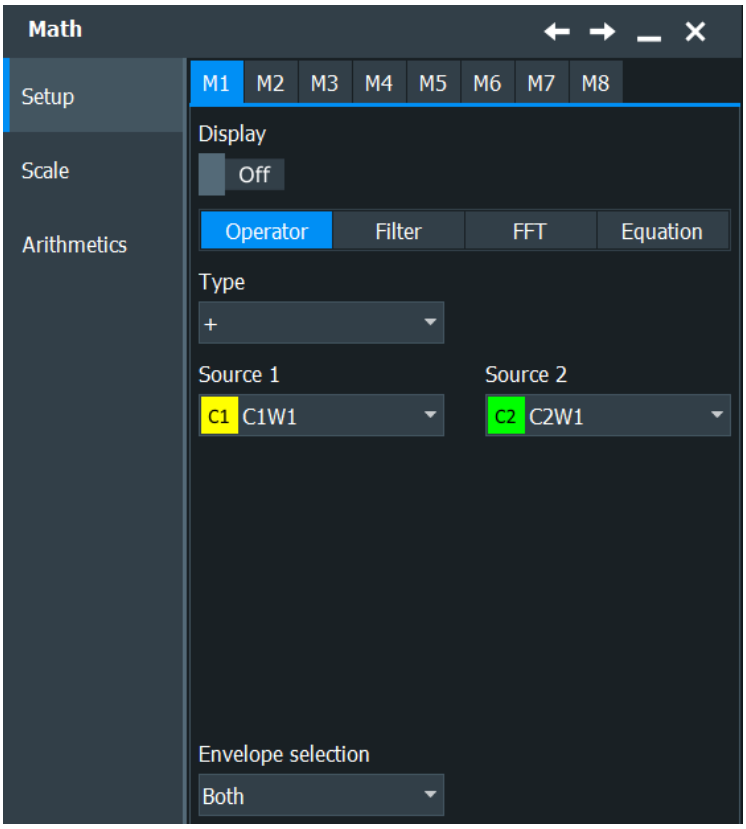
Envelope selection
Selects the upper or lower part of the waveform for mathematic calculation, or a combination of both.

The setting is relevant for waveforms with waveform arithmetic mode "Envelope" or with "Peak detect" decimation. All mathematic operations - except for derivation - can be applied to envelope waveforms and waveforms with "Peak detect" decimation.

Remote command:
`CALCulate:MATH<m>:ENVSelection` on page 1457

7.3.3 Operator editor

In the Operator editor, you can define the most common mathematical formulas without knowing their correct syntax.



Remote command:

- `CALCulate:MATH<m>[:EXPRession][:DEFine]` on page 1456

7.3.3.1 Settings in the Operator editor

Source 1 / 2 260
Type 260
Noise reject 260
a / b 260

Source 1 / 2
Defines the signal source to be evaluated by the math function. Waveform 1 of channel signals can be selected.

Note: If you require other signal sources not listed here, use the formula editor provided in the Equation tab. Any waveform of any input channel can be used as a source. See: Chapter 7.3.4, "Equation", on page 261.

Type
Defines the type of operation to be performed on the selected signal sources. The following functions are available:

Note: If you require other operators not listed here, use the formula editor provided in the "Equation" tab. See: Chapter 7.3.4, "Equation", on page 261.

"+"	Adds up the sources
"-"	Subtracts source 2 from source 1.
"x"	Multiplies source 1 by source 2.
"|x|"	Determines the absolute value of the source.
"dx/dt"	Differentiates the source value with respect to the time value. Not possible on envelope waveforms and waveforms with "Peak detect" decimation.
"log(x)"	Calculates the logarithm of the source value based on 10.
"ln(x)"	Calculates the natural logarithm of the source value (based on e).
"ld(x)"	Calculates the binary logarithm of the source value (binary logarithm, based on 2).
"Rescale"	Rescales the source values by a factor "a" and an offset "b": *ax+b*. See also: "a / b" on page 260.

Noise reject
Only available for "Type" = "dx/dt".

Sets the number of neighboring samples that are skipped for differentiation.

To suppress noise effects during differentiation, it can be useful not to consider two directly neighboring points to calculate *dx* (x_n-x_{n-1}). Instead, some samples in-between are skipped and a point a few samples further is used (e.g. x_n-x_{n-3}).

a / b
Defines the values for the "Rescale" function (ax+b).

"a"	Multiplication factor
"b"	Offset of the signal source on the y-axis.

7.3.3.2 Defining a formula in the Operator editor

1. Open the "Menu" > "Math" dialog.
   Alternatively, press the [Math] key.
2. In the "Setup" tab, select the "Operator" tab.

3. Tap the "Source 1" and "Source 2" icons and select the signal sources to which the math function is applied. For details on available signal sources, see "Source 1 / 2" on page 260.
4. Tap the "Type" icon and select the mathematical function.
   For details on available operators, see "Type" on page 260.
5. If the operator requires additional parameters, enter them in the input fields.

7.3.4 Equation

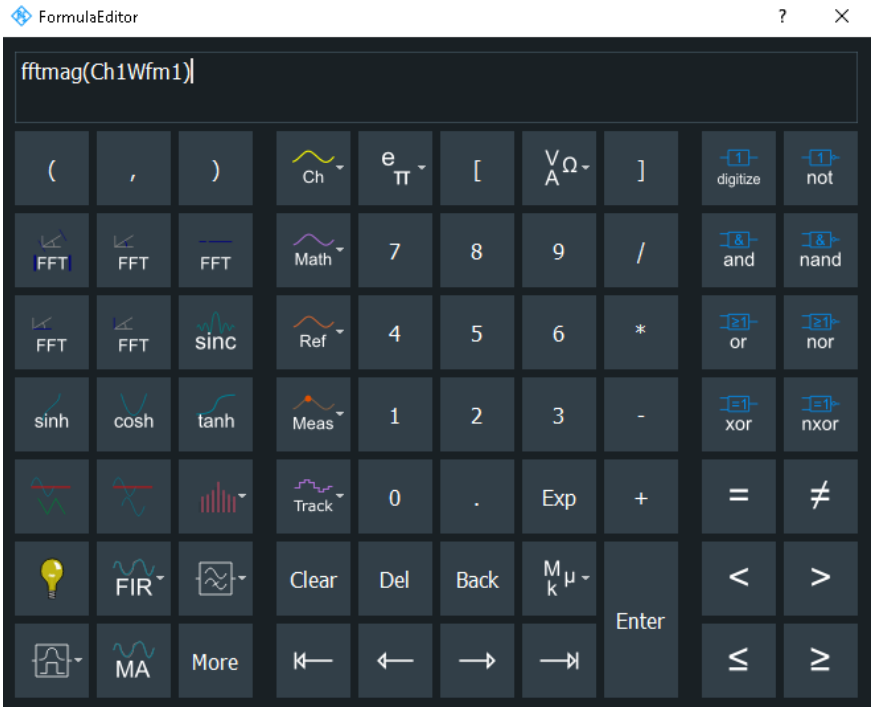
In the Equation tab, you can enter complex formulas to define a math waveform. The formula editor helps to enter formulas easily with correct syntax, using a large selection of operators and signal sources.

All formulas in the formula editor are linear regardless whether a [dB] is set or not.

► Double-tap the Equation tab to display the formula editor.

7.3.4.1 Formula editor

Using the formula editor you can define math functions freely, using a large selection of operators and signal sources. For a procedure on using the editor, see Chapter 7.3.4.11, "Defining a formula in the formula editor", on page 268.



Remote command:

- `CALCulate:MATH<m>[:EXPRession][:DEFine]` on page 1456

Buttons of the formula editor and their usage

Table 7-1: Basics

Icon	Description	Usage/Comment, *FormulaEditor expression*
(	left bracket	enclose operands
,	comma	separates operands
)	right bracket	enclose operands
e / π	math. constants	e: Euler number: 2.7182... Pi: 3.1415...
[	left square bracket	enclose unit
V / A / Ω	units	[<unit>]
]	right square bracket	enclose unit
x^a	exponentiation with base x	x: base, a: exponent *x^a*
/	division	
*	multiplication	
-	subtraction	
+	addition	
0...9	numeric characters	
.	decimal point	
Exp	exponentiation with base 10	*e*
Enter	expression complete	insert expression in "Setup" dialog and close the formula editor
Clear	clear expression in editor	restart editing
Del	Delete	remove selected part of expression
Back	Backspace	remove last symbol, operator or operand to the left of the cursor
M / k / µ	SI-prefix for unit	<SI-prefix>[<unit>]

Table 7-2: Signal sources

Icon	Description	Usage/Comment, *FormulaEditor expression*
Ch	signal waveform	*Ch<1...4>Wfm<1...3>*
Math	math waveform	*Math<1...8>*
Ref	reference waveform	*Ref<1...4>*

Icon	Description	Usage/Comment, *FormulaEditor expression*
Meas	measurement waveform	*Meas<1...8>*
Track	track waveform	*Track<1...8>*

Table 7-3: Cursor keys

Icon	Description	Usage/Comment, *FormulaEditor expression*
|←	move cursor to beginning	
←	move cursor 1 step to the left	
→	move cursor 1 step to the right	
→|	move cursor to end	

7.3.4.2 Math functions: algebra

Table 7-4: Algebra

Icon	Description	Usage/Comment, *FormulaEditor expression*
|x|	absolute x value	*abs(x)*
√x	square root of x	*sqrt(x)*
x^2	x*x	*pow(x)*
log_{10}	common logarithm (base 10)	*log(x)*
log_e	natural logarithm (base e)	*ln(x)*
log_2	binary logarithm (base 2)	*ld(x)*
e^x	exponentiation with base e	*exp(x)*
∫xdx	integral of x	*integral(x)*
d/dx	derivation of x	*derivation(x,y)* with x = waveform and y = number of skipped samples (noise reject)
ax+b	scaling of x	*rescale(x,a,b)*

7.3.4.3 Math functions: trigonometry

Table 7-5: Trigonometry (More keys)

Icon	Description	Usage/Comment, *FormulaEditor expression*
sinh	hyperbolic sine	*sinh(x)*
cosh	hyperbolic cosine	*cosh(x)*
tanh	hyperbolic tangent	*tanh(x)*

7.3.4.4 Math functions: bit operations

Table 7-6: Bit operations

Icon	Description	Usage/Comment, *FormulaEditor expression*
digitize	convert to 0 or 1	*digitize(x)* The function uses the middle reference level and the hysteresis, which are set in "Menu" > "Measure" > "Advanced" tab > "Reference Level".
not	negation	*not(x)*
and		*and*
nand	negation of and	*nand*
or		*or*
nor	negation of or	*nor*
xor	exclusive or	*xor*
nxor	negation of exclusive or	*nxor*

7.3.4.5 Math functions: comparison

Table 7-7: Comparison

Icon	Description	Usage/Comment, *FormulaEditor expression*
=	equal	=
≠≠	not equal	<>
<	smaller	<
>	greater	>
≤	smaller or equal	<=
≥	greater or equal	>=
More	display additional keys	

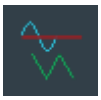
7.3.4.6 Math functions: FFT

Table 7-8: FFT (More keys)

Icon	Description	Usage/Comment, *FormulaEditor expression*
|FFT|	magnitude of FFT value	*fftmag(x)*
FFT (φ)	FFT phase value	*fftphi(x)*
FFT -dφ*df	FFT group delay	*fftgroupdelay(x)*
FFT (re)	real part of FFT value	*fftre(x)*
FFT (im)	imag part of FFT value	*fftim(x)*

7.3.4.7 Math functions: correlation

Table 7-9: Correlation (More keys)

Icon	Description	Usage/Comment, *FormulaEditor expression*
	Cross correlation function of two waveforms Measures the similarity of two waveforms as a function of a time-lag applied to one of them. Function limits the maximum record length to 4 MSa. Two modes of normalization are supported: biased and unbiased. The length of the correlation buffer is $N_0 + N_1 - 1$ samples. The length of the first input signal is N_0 samples and the length of the second signal is N_1 samples.	*correlation(x1, x2, biased)* *correlation(x1, x2, unbiased)* with x1 = waveform 1 and x2 = waveform 2 *correlation(x1, x2)* performs an unbiased correlation
	Auto correlation Used to find repeating patterns, for example, a periodic signal obscured by noise. The length of the auto correlation buffer is $2N - 1$ samples, if the length of the input signal is N samples. Two modes of normalization are supported: biased and unbiased.	*autocorrelation(x, biased)* *autocorrelation(x, unbiased)* with x = channel waveform *autocorrelation(x)* performs an unbiased autocorrelation
	biased / unbiased normalization for correlation and auto correlation	see above

Mathematic background for correlation:

$$Temp1_R_{xy}(m) = \sum_{n=0}^{N1} y_n^* x_{n+m} \qquad m \in [0; N_1[$$

$$Temp0_R_{xy}(m) = \sum_{n=1}^{N0} x_n^* y_{n+m} \qquad m \in [1; N_0[$$

The R&S RTO6 uses only the real part of the signal. Two modes of normalization are supported: biased and unbiased.

$$R_{xy}(m) = \begin{cases} \dfrac{1}{\min(N_0, N_1)} Temp1_R_{xy}(m) & m \in [N_0 - 1; N_1 + N_0 - 1[\\ \dfrac{1}{\min(N_0, N_1)} Temp0_R_{xy}^*(-m) & m \in [0; N_0 - 1[\end{cases}$$

Equation 7-1: Biased correlation

$$R_{xy}(m) = \begin{cases} \dfrac{1}{a(m)} Temp1_R_{xy}(m) & m \in [N_0 - 1; N_1 + N_0 - 1[\\ \dfrac{1}{a(m)} Temp0_R_{xy}^*(-m) & m \in [0; N_0 - 1[\end{cases}$$

Equation 7-2: Unbiased correlation

Mathematic background for auto correlation:

$$R_{xx}(m) = \begin{cases} \frac{1}{N} \sum_{n=0}^{N-m-1} x_n^* x_{n+m} & m \geq 0 \\ R_{xx}^*(-m) & m < 0 \end{cases}$$

Equation 7-3: Biased auto correlation

$$R_{xx}(m) = \begin{cases} \frac{1}{N-|m|} \sum_{n=0}^{N-m-1} x_n^* x_{n+m} & m \geq 0 \\ R_{xx}^*(-m) & m < 0 \end{cases}$$

Equation 7-4: Unbiased auto correlation

The R&S RTO6 uses only the real part of the signal.

7.3.4.8 Math functions: filter and power

Table 7-10: Filter and power (More keys)

Icon	Description	Usage, comment, *FormulaEditor expression*
	Electric power	Electric power is calculated from voltage, based on measurement impedance (see Impedance) *elecpower(x)* = U^2/R
	Finite impulse response (FIR) filter	*FIR(tpye,source,limit,shape)* or *FIR(userdef,source,path)* See Chapter 7.3.5.2, "FIR filter in the formula editor", on page 270
	Type of FIR filter	*highpass* or *lowpass*, see FIR filter
	Characteristics of FIR filter	*gaussian* or *rectangle*, see FIR filter
	Moving average	Calculates a mean value of several adjacent sample points. The result is a smoothed waveform. The moving average uses the full data and can be used for non-periodic signals. It works like a lowpass filter and increases the vertical resolution at the expense of bandwidth reduction. *MovingAverage(x,y)* with: x = source (channels only), y = number of samples to be averaged Example: *MovingAverage(Ch1Wfm1,1000)* Averages 1000 subsequent samples of the channel 1 waveform

7.3.4.9 Math functions: CDR

The clock data recovery function requires at least one of the options R&S RTO6-K12, or R&S RTO6-K133, or R&S RTO6-K134.

Hardware CDR requires option R&S RTO6--K13.

Table 7-11: Clock data recovery (More keys)

Icon	Description	Usage/Comment, *FormulaEditor expression*
	CDR: displays the generated clock signal as math waveform	Hardware: *CDR(hw)* Software 1: *CDR(sw1,x)* Software 2: *CDR(sw2,x)* where x is the signal from which the clock is recovered Example: *CDR(sw1,CH2Wfm1)* See also: Chapter 18.3.3, "Displaying the recovered clock signal", on page 1120

7.3.4.10 Transfer functions of a step-like signal

The R&S RTO6 provides two functions to calculate the transfer function of a system out of its step response. These functions are not supported by the formula editor, but you can enter them directly in the "Advanced" tab in the "Math Setup" dialog box. To enter the formula, you need a keyboard (onscreen keyboard or a connected usual one).

Math expression	Description
Step2FreqRespNormMag(x,points) Where: • x is the waveform (channel, math, reference) • points is the number of points in the resulting math waveform (min. 1000 points)	Normalized magnitude of transfer function in frequency domain Example: *Step2FreqRespNormMag(Ch1,2000)*
Step2FreqRespNormPhi(x,points,time offset) Where: • x is the waveform (channel, math, reference) • points is the number of points in the resulting math waveform (min. 1000 points) • time offset in s	Normalized phase of transfer function in frequency domain Example: *Step2FreqRespNormPhi(Ch1,2000,2e-9)*

The calculation of transfer functions has several steps:

- Calculate the frequency domain representation of a step-like signal. A detailed description is given in: A. M. Nicolson, „Forming the fast Fourier Transform of a step response in Time-Domain Metrology,“ in Electronic letters, Bd. 9, Nr. 14, p. 317, 1973.
- Calculate the derivative to convert the result to the transfer function.
- Normalize the result at DC to a magnitude of 1 V and subtract specified time offset from phase.

The step-like signal to be analyzed is a time domain waveform, and the result is a math waveform in the frequency domain.

To get useful results, check and adjust the following settings:

1. Place the analyzed step of the signal in the left half of the time domain diagram. Use the "Reference point" or the horizontal "Position" of the trigger to move the signal.
   See also: Chapter 5.4, "Acquire settings", on page 126.
2. Adjust the record lengths of the source signal and the math signal. The "Record length" of the step-like source signal must be shorter than the number of points in the resulting math function, which is set in the formula. See also: "Record length" on page 127.
3. If the resulting math waveform is noisy, increase the "Average count" for the source signal until the waveform is clear.
   See also: Chapter 5.4.2, "Mode settings", on page 129.

7.3.4.11 Defining a formula in the formula editor

1. Open the "Menu" > "Math" dialog.
2. Select the Equation tab.
3. Double-tap the editing area.

   The "Formula Editor" is displayed.
4. Enter the math formula including all required signal sources and operators by selecting the corresponding keys in the editor. For details on the available keys, see Chapter 7.3.4.1, "Formula editor", on page 261.
5. To insert a physical unit in the formula, proceed as follows:
   a) If necessary, insert a decimal prefix using the "M/k/µ" key.
   b) Insert an opening square bracket using the "[" key.
   c) Insert the physical unit using the "V/A/Ω" key.
   d) Insert a closing square bracket using the "]" key.

   The resulting expression could be, for example: `m[V]`
6. To perform a rescaling function, proceed as follows:
   a) Select the rescaling function using the "ax+b" key.
   b) Behind the left bracket, insert the signal source using one of the following keys:
      - "Ch" for a channel
      - "Math" for a math function
      - "Ref" for a reference waveform
      - "Meas" for a measurement
   c) Insert a comma using the "," key.
   d) Insert the "a" value, i.e. the scaling factor, using the number keys.

e) Insert a comma using the "," key.
f) Insert the "b" value, i.e. the scaling offset, using the number keys.
g) Insert the closing bracket using the ")" key.

The resulting expression could be, for example: `rescale(Ch1Wfm1,3,4)`

7.3.5 Filters

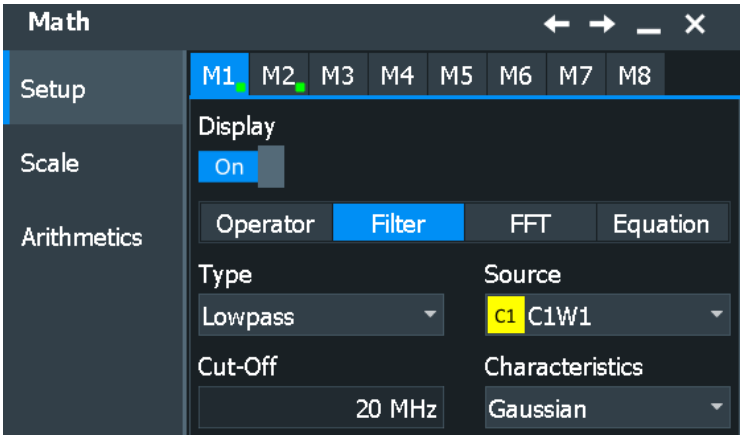
The R&S RTO6 provides several ways to filter the input signal:

- Lowpass filter by selecting a bandwidth limit in the acquisition path (vertical channel settings)
  See "Analog" on page 141
- Digital lowpass filter
  See Chapter 5.5.5, "Digital filter setup", on page 143
- FIR filter to create a filtered math waveform (lowpass, highpass, bandpass)
  You can set up a FIR-filtered math waveform in the basic math editor, or in the advanced editor. Both ways are described below in this chapter.

7.3.5.1 FIR filter in the math setup

Access: "Math" > "Setup" tab > "Filter" tab.

The finite impulse response filter is a filter to create filtered waveforms with lowpass, highpass, or bandpass. The filter requires additional settings.



- "Type": defines whether the FIR filter is a highpass, lowpass, or bandpass/bandstop (= "User defined") filter.
- "Cut-Off": sets the limit frequency for the highpass or lowpass FIR filter.
- "Characteristics": relevant for lowpass filter. Defines whether it has a Gaussian or a rectangular shape. The highpass is always Gaussian.
- "Select filter file": relevant for bandpass/bandstop filter. Opens a file dialog to select the file with the filter coefficients.

Cut-off frequency for lowpass FIR filter

The cut-off frequency depends on the horizontal resolution and the filter characteristics. The frequency for the lowpass filter can only be set in this range:

f_g_3dB = (0.001 ... 0.2) f_a_in* for Gaussian FIR filter

f_g_3dB = (0,001 ... 0.4) f_a_in* for rectangular FIR filter

Where: *f_g_3dB* = cut-off frequency to be set for the lowpass filter, and *f_a_in* = reciprocal of the resolution, or sample rate.

Cut-off frequency for highpass FIR filter

To check limit frequency for the highpass filter, convert it to an equivalent lowpass frequency:

f_LP = f_a_in/2 - f_HP

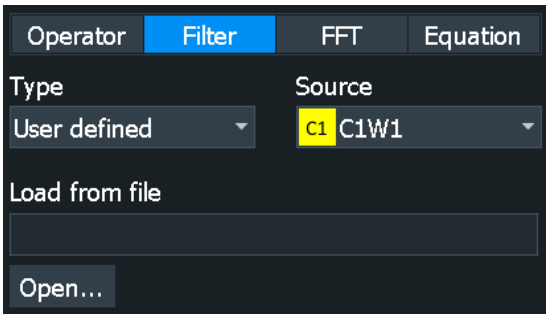
Where f_HP is the requested highpass limit frequency and f_LP the equivalent lowpass frequency that has to comply with the limits given above.

Bandpass and bandstop FIR filter: CSV file

To define a bandpass or bandstop, you need a CSV file that contains the comma-separated filter coefficients. The maximum number of filter coefficients is 2 048 000 taps. No other parameters are allowed in the file.

To create the CSV file, we recommend using the Matlab Filter Design & Analysis tool. In the tool, enter the filter type, filter order and filter frequencies. Make sure to set the sample frequency in Matlab and the sample rate at the oscilloscope to the same value. If the values differ, the filter is shifted in frequency.

In R&S RTO6, in the "Math" > "Setup" > "Filter", select the filter "Type" = "User defined", and load the CSV file.



7.3.5.2 FIR filter in the formula editor

You can type the FIR filter formula directly in the "Equation" tab, or open the formula editor and use the buttons to create the formula.

Highpass and lowpass FIR filter

The general syntax for highpass and lowpass filters is:

FIR(tpye,source,limit,shape)

- Type is *lowpass* or *highpass*
- Source is the input channel of the signal.
- Limit is the cut-off frequency. Consider the range of possible values, see "Cut-off frequency for lowpass FIR filter" on page 270.
- Shape is *gaussian* or *rectangle* for the lowpass filter, and *gaussian* for the high-pass.

► For example, to set a rectangle lowpass filter on channel 1 with 10 MHz cut-off frequency, enter:
FIR(lowpass,Ch1,1e+07,rectangle)

► For example, to set a Gaussian highpass filter on channel 2 with 3 GHz cut-off frequency, enter:
FIR(highpass,Ch2,3e+09,gaussian)

Bandpass and bandstop FIR filter

The syntax for the bandpass filter is:

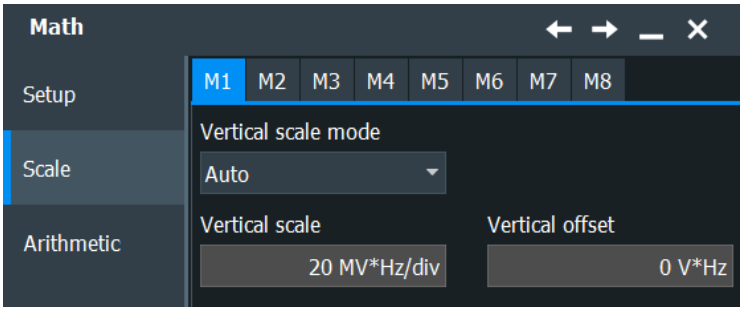
FIR(userdef,source,path)

- Source is the input channel of the signal.
- Path is a string containing the path and filename of the filter file. The file contains the comma-separated filter coefficients.

► For example, to set a bandpass on channel 1 with filter coefficients saved in the `bandpass.csv` file, enter:
FIR(userdef,Ch1,"C:\Users\Public\Documents\Rohde-Schwarz\RTx\bandpass.csv")

7.3.6 Math scale settings

Access: [Math] > "Scale" tab.



In this tab, you can find functions to set the vertical parameters of the math waveform.

For FFT scale settings, see Chapter 9.1.4, "FFT scale settings", on page 387.

Vertical scaling mode (Manual/Auto)
By default, the vertical scale is adapted to the current measurement results automatically to provide an optimal display. However, if necessary, you can define scaling values manually to suit your requirements.

Note: When you change the scaling values manually using the "Scale" rotary knob, the scale mode is set to "Manual" temporarily. When you edit the math function, scaling is automatically set back to "Auto" mode. "Manual" mode is only maintained during math function changes if you select it yourself.

"Manual"	Enter the required values for "Vertical scale" and "Vertical offset". For FFT, set "Vertical range" and "Vertical maximum".
"Auto"	"Vertical scale" and "Vertical offset" are read-only. For FFT, only the "Vertical maximum" is read-only.

Vertical Scale
Defines the scale of the y-axis in the math function diagram. The value is defined as "<unit> per division", e.g. *50m V/div*. In this case, the horizontal grid lines are displayed in intervals of 50 mV.

If the "Vertical scaling mode (Manual/Auto)" on page 272 is set to "Auto", this setting is read-only.

Remote command:
`CALCulate:MATH<m>:VERTical:SCALe` on page 1459

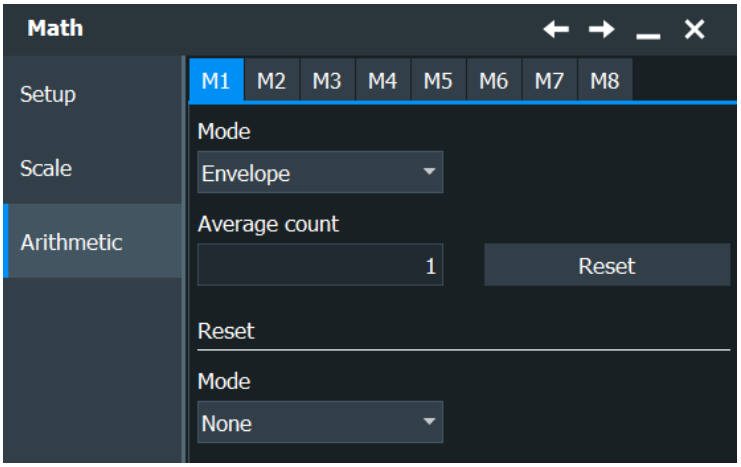
Vertical Offset
Sets a voltage offset to adjust the vertical position of the math function on the screen. Negative values move the waveform, positive values move it down.

If the "Vertical scaling mode (Manual/Auto)" on page 272 is set to "Auto", this setting is read-only.

Remote command:
`CALCulate:MATH<m>:VERTical:OFFSet` on page 1458

7.3.7 Math arithmetic

Access: [Math] > "Arithmetic" tab.



In this tab you can specify the waveform arithmetic for the math waveforms.

Mode

Waveform arithmetic builds the resulting waveform from several consecutive acquisitions and subsequent math calculations of the signal. For details, see "Arithmetic" on page 130.

"Original"	The original results are displayed.
"Envelope"	The envelope curve of all acquired and calculated results is displayed.
"Average"	The average of all acquired and calculated results is displayed.
"RMS"	The root mean square of the math data is displayed. The result is the average power spectrum. If you measure the channel power on this RMS spectrum, you get the same result as for the average channel power measurement on waveforms.
"MinHold"	Determines the minimum result for each input value from the data of the current acquisition and some acquisitions before.
"MaxHold"	Determines the maximum result for each input value from the data of the current acquisition and some acquisitions before.

Remote command:

`CALCulate:MATH<m>:ARIThmetics` on page 1457

Average count (N-single count)

Access:

- "Menu" > "Acquire" > "Setup" tab > "N-single count"
- "Menu" > "Acquire" > "Mode" tab > "Average count"
- "Menu" > "Acquire" > "Segmented" tab > disable "Acquire maximum" > "Required"
- [Math] > "Arithmetic" tab > "Mode" is not "Off" > "Average count"

The acquisition and average count has several effects:

- It sets the number of waveforms acquired with [Single]
- It defines the number of waveforms used to calculate the average waveform. Thus, the instrument acquires sufficient waveforms to calculate the correct average if "Average" is enabled for waveform arithmetic. The higher the value is, the better the noise is reduced.

- It sets the number of acquisitions to be acquired in a fast segmentation acquisition series. Thus, you can acquire exactly one fast segmentation acquisition series with [Single].
  If fast segmentation is enabled and configured to acquire the maximum number of acquisitions, the acquisition count is set to that maximum number and cannot be changed.
  See also "Number of acquisitions" on page 132.
- It is the "Finished" criteria for the state of a mask test.

Remote command:
`ACQuire:COUNt` on page 1334

Reset
Forces the immediate restart of the envelope and average calculation for all waveforms.

Remote command:
`ACQuire:ARESet:IMMediate` on page 1335

Arithmetic Reset > Mode
Defines when the envelope and average evaluation restarts.

"None"	No restart, the number of acquisitions considered by the waveform arithmetics is not limited.
"Time"	Restarts the envelope and average calculation after the time defined in "Time".
"Waveforms"	Restarts the envelope and average calculation after a number of acquired waveforms defined in "Count".

Remote command:
`ACQuire:ARESet:MODE` on page 1335
`ACQuire:ARESet:TIME` on page 1335
`ACQuire:ARESet:COUNt` on page 1335

7.4 History

The history accesses the data of previous acquisitions and provides them for further analysis.

7.4.1 About history

If a continuous acquisition runs, the captured data is stored in the sample memory and the current acquisition is processed and shown on the display. After the acquisition was stopped, the history accesses the captured samples that were stored, displays these samples as history waveforms, and makes them available for further analysis. It considers all channels that were enabled during the running acquisition. When a new acquisition is started with [Run Stop] or [Single], the memory is cleared and written anew.

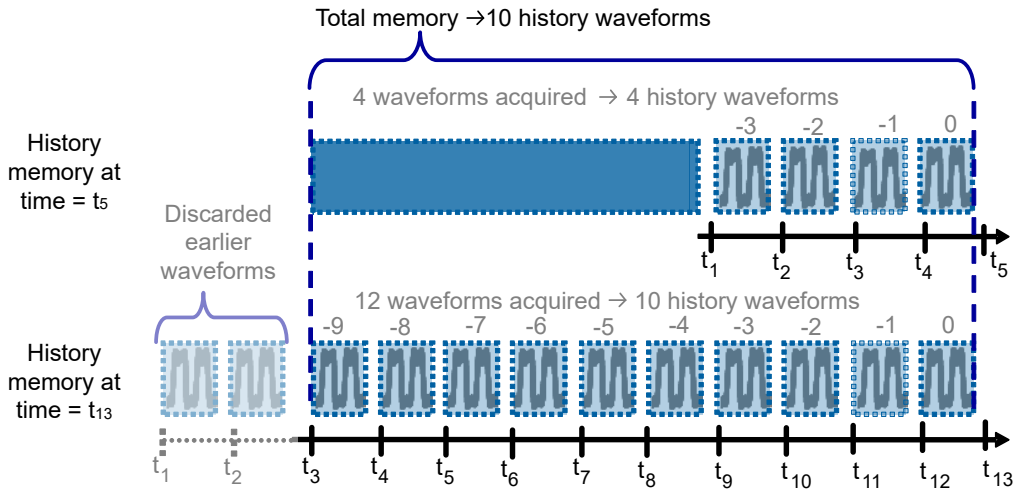


Figure 7-5: History memory. In this example, the memory can store 10 waveforms.

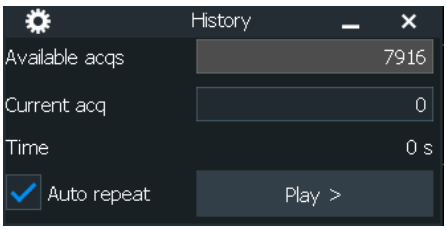
You can work with history waveforms in the same way as with the waveform of the latest acquisition: use zoom, cursor measurements, and automatic measurements, create math waveforms, perform mask testing and so on. Saving the history data is also possible, either completely or a part of the data.

The number of stored history waveforms depends on the memory size, the number of enabled channels, and the record length. The shorter the record length, the less the number of channels, and the larger the memory, the more history waveforms are saved.

Quick-access History dialog box

When you press the [History] key on the front panel or tap "Display" menu > "Show history", the history mode is enabled and the quick-access "History" dialog box is displayed. A running acquistion stops immediately.

The small quick-access "History" dialog box can remain visible on the screen during history replay, so that the history can be replayed at any time by a simple tap on the "Play" button. Closing the quick-access "History" dialog box, or starting a new acquisition disables the history mode.



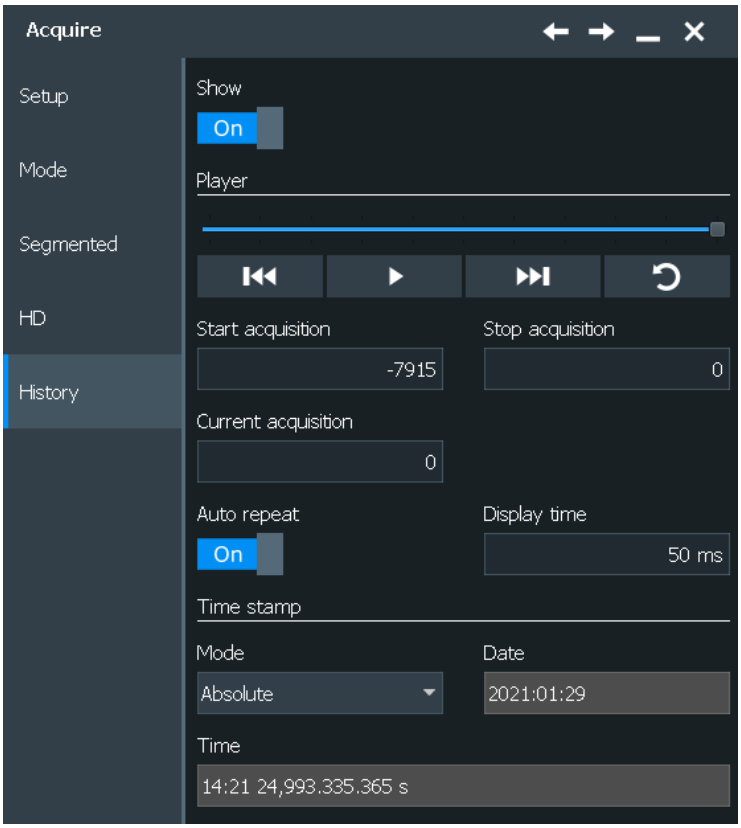
Export of history waveforms

7.4.2 History setup

The "History" dialog contains the complete functionality on history viewing and information.

The most important information and functions are also provided in the quick-access history dialog box.

Access: "Menu" > "Acquire" > "History" tab.



Show history / Export history
Enables the history mode and allows you to save history waveforms to file.

The history display is enabled automatically when you press the [History] button. It is disabled when you close the quick-access "History" dialog box.

For details on data export, see Chapter 12.2.2.2, "Waveform history settings", on page 463.

Remote command:
`CHANnel<m>[:WAVeform<n>]:HISTory[:STATe]` on page 1461

Player
The player can be used to control the playback of the history waveforms.

⏮	Sets the oldest acquisition in the sample memory as "Start acquisition" and "Current acquisition".
▶	Starts and stops the replay of the history waveforms from "Start acquisition" to "Stop acquisition".
⏭	Sets the newest acquisition in the sample memory as "Stop acquisition" and "Current acquisition". The newest acquisition always has the index "0".
↺	Resets the "Stop acquisition" to 0 and the "Start acquisition" to the number of saved acqusitions. Thus, the history viewer shows all acquisitions.

Start acquisition
Sets the index of the first (oldest) acquisition to be displayed or exported. The index is always negative.

Remote command:
`CHANnel<m>[:WAVeform<n>]:HISTory:STARt` on page 1462

Stop acquisition
Sets the index of the last (newest) acquisition to be displayed or exported. The newest acquisition of the complete acquisition series always has the index "0".

Remote command:
`CHANnel<m>[:WAVeform<n>]:HISTory:STOP` on page 1462

Current acquisition
Accesses a particular acquisition in the memory to display it, or to save it. The newest acquisition always has the index "0". Older acquisition have a negative index.

If a history replay is running, the field shows the number of the currently shown acquisition.

Remote command:
`CHANnel<m>[:WAVeform<n>]:HISTory:CURRent` on page 1461

Auto repeat
If selected, the replay of the history waveform sequence repeats automatically. Otherwise, the replay stops at the "Stop acquisition".

Remote command:
`CHANnel<m>[:WAVeform<n>]:HISTory:REPLay` on page 1463

Display time
Sets the display time for one acquisition. The shorter the time, the faster the replay is.

The setting takes effect for history replay and the display of a Fast Segmentation series, see Chapter 5.4.3, "Segmented settings", on page 132.

Remote command:
`CHANnel<m>[:WAVeform<n>]:HISTory:TPACq` on page 1463

Time stamp

The time stamp shows the time of the currently displayed history acquisition. Thus, the time relation between acquisitions is always available.

The time stamp "Mode" can be absolute or relative:

- In "Absolute" mode, the instrument shows the date and the daytime of the current acquisition.
- In "Relative" mode, the time difference to the newest acquisition (index = 0) is shown.

The time stamp can be included in waveform data export, see"Time stamps" on page 463.

During history replay, the time value is displayed and updated if the replay speed ("Display time") is slow enough, that is 40 ms or slower.

The quick-access history dialog box always shows the relative time. In the "History Viewer" tab, you can select the time mode.

Remote command:

`CHANnel<m>[:WAVeform<n>]:HISTory:TSDate?` on page 1464
`CHANnel<m>[:WAVeform<n>]:HISTory:TSABsolute?` on page 1464
`CHANnel<m>[:WAVeform<n>]:HISTory:TSRelative?` on page 1464
`CHANnel<m>[:WAVeform<n>]:HISTory:TSRReference?` on page 1465

7.4.3 Using history

You can access the history waveforms in two ways:

- Display a particular acquisition.
- Replay all or a part of the saved waveforms to track the signal run.

Furthermore, you can export history data to a file.

- "To open the history and get information" on page 278
- "To display a particular acquisition" on page 279
- "To replay history waveforms" on page 279
- "To exit the history" on page 279
- "To save the history data" on page 280

To open the history and get information

1. Press the [History] key on the front panel.

   A running acquisition is stopped, the history mode is enabled and the quick-access "History" dialog box is displayed. The [History] key lights up as long as the history mode is active.

2. Open the full configuration dialog box:
   - Tap the icon.
   - Open "Menu" > "Acquire" dialog > "History" tab.

To display a particular acquisition

1. In the quick-access "History" dialog box, enter the number of the required acquisition in the "Current acq" field. The newest acquisition always has the index "0", older acquisitions have a negative index.
2. Tap "Play" to start.

Alternatively, you can configure and start the history display from the "History" configuration dialog box:

1. Open the "History" configuration tab.
2. If the history mode is off (the [History] key is not illuminated), select "Show".

   The quick-access dialog box is displayed.
3. Drag the slider to the required acquisition. The current number is shown in the "Current acquisition" field.
   Alternatively, enter the number of the required acquisition in the "Current acquisition" field.
4. Tap "Play" to start.

To replay history waveforms

If you want to see the complete acquisition series without any setup, simply tap "Play" in the quick-access "History" dialog box. For specific analysis of history data, use the history "Viewer" setup.

1. Open the "History" configuration dialog tab.
2. If the history mode is off (the [History] key is not illuminated), enable "Show".

   The quick-access dialog box is displayed.
3. Define the part of the history you want to see by doing one of the following:
   - Tap to see the complete history.
   - Enter the "Start acquisition" of the oldest acquisition to display and the "Stop acquisition" of the newest acquisition to display. All waveforms between the two indexes are displayed.
     To enter the oldest or newest acquisition for either index, tap the appropriate button. The newest acquisition always has the index "0". The "Start acquisition" is always negative.
4. Tap to start.

To exit the history

► Choose one of the following ways:
- Close the quick-access "History" dialog box.
- In the "History" configuration tab, disable "Show".
- Start the acquisition.

To save the history data

You can save the complete history, or some subsequent waveforms from the history, or a single history waveform. You can also decide to save the complete waveforms, or a part of each waveform.

1. Open the "Menu" > "Save/Recall" dialog.
2. Select the "Save/Recall" tab.
3. In the "Save " tab, press "Waveform".
4. In the "Setup" tab, tap the source icon to select the waveform you want to save.
5. If you want to save only a part of each waveform, set the "Scope".
   For settings, see "Scope" on page 461.
6. Select the "History" tab.
7. Enable "Export history".
8. If you want to write the timestamps into the data file, enable "Time stamps".
9. To save one waveform out of the history memory:
   a) Make sure that "Multiple acquisitions in one file" is disabled.
   b) Enter the number of the required acquisition in "Acquistion index".
      The newest acquisition in the memory always has the index "0". Older acquisitions have a negative index.
   c) Tap "Save" or "Save As" to save the waveform data to the specified file.
10. To save several subsequent history waveforms:
    a) Enable "Multiple acquisitions in one file".
    b) Define the range of the waveforms to be saved with "Start acquisition" and "Stop acquisition".
    c) Tap "Start export" to play the history and to save the history data to the specified file.

Se also Chapter 12.2.2.2, "Waveform history settings", on page 463.

7.5 XY-diagram

XY-diagrams combine the voltage levels of two waveforms in one diagram. They use the voltage level of a second waveform as the x-axis, rather than a time base. This allows you to perform phase shift measurements, for example. You can display up to four different XY-diagrams.

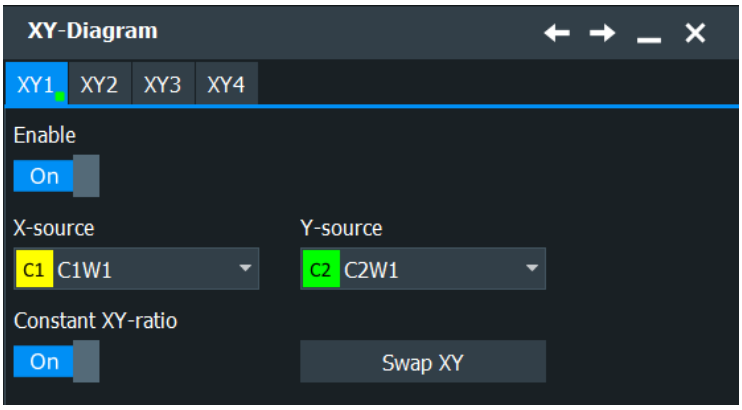
XY-diagrams can be used to display the IQ representation of a signal.

7.5.1 Settings for XY-diagrams

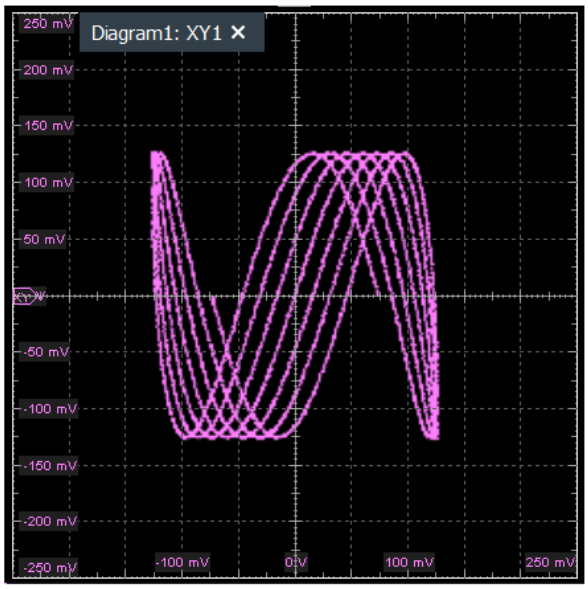
Access: [App Cockpit]>"Analysis" tab > "XY-Diagram"

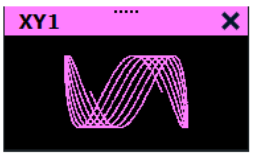
You can display up to four different XY-diagrams that use the voltage level of a waveform as the x-axis, rather than a time base.

Make sure to select the tab of the required XY-diagram.



Enable
If activated, the XY-waveform is active and shown in a diagram, or it is minimized in a signal icon.





Remote command:
`WAVeform<m>:XYCurve:STATe` on page 1466

X-source
Defines the signal source that supplies the x-values of the XY-diagram. Select one of the following:

- One of the waveforms of any channel
- A reference waveform

- The results of a mathematical function

Remote command:
`WAVeform<m>:XYCurve:XSOurce` on page 1467

Y-source
Defines the source to be used as the y-axis of the XY-diagram. Select one of the following:

- One of the waveforms of any channel
- A reference waveform
- The results of a mathematical function

Remote command:
`WAVeform<m>:XYCurve:YSOurce` on page 1467

Constant XY-ratio
If enabled, the x- and y-axes maintain a constant ratio in the diagram.

Remote command:
`WAVeform<m>:XYCurve:RATio` on page 1465

Swap XY
Replaces the source of the x-axis with the source of the y-axis and vice versa.

Remote command:
`WAVeform<m>:XYCurve:SWAP` on page 1466

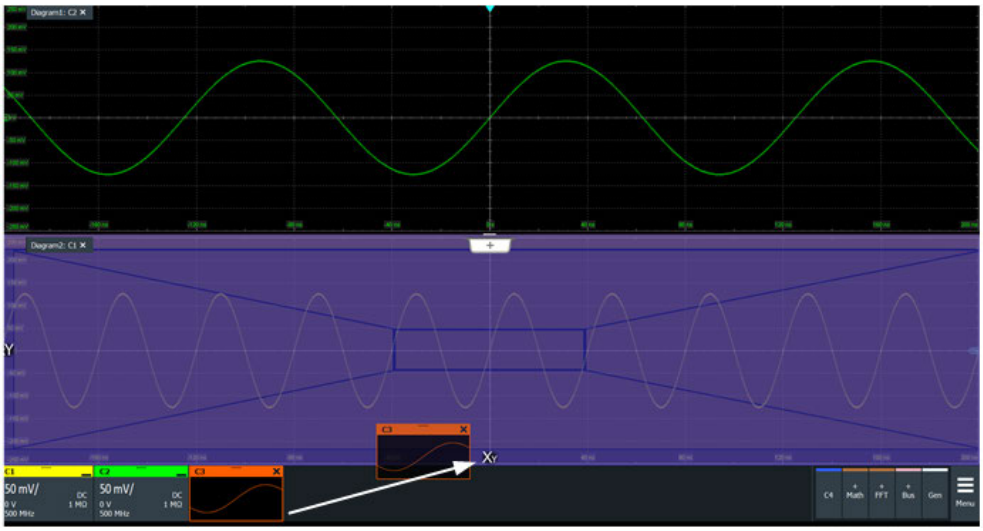
7.5.2 Displaying an XY-diagram

You can create the diagram from active waveforms with drag&drop, or use the dialog box for setup.

To display an XY-diagram with drag&drop

Prerequisites: The source waveform for the y-axis is active in a diagram, the source waveform for the x-axis is either active or minimized.

1. Drag the x-axis waveform to the lower middle of the diagram with the y-axis waveform.
2. Drop the icon when it overlaps with the Xy shown in the middle of the diagram.



The diagram is converted into an XY-diagram.

To set up an XY-diagram

1. Press the [App Cockpit] key.
2. In the >"Analysis" tab, tap the "XY-Diagram".
3. Enable the "XY-diagram".
4. In the "X-source" field, define the signal source that supplies the x-values of the XY-diagram. Select one of the following:
   - One of the waveforms of any channel
   - A reference waveform
   - The results of a mathematical function
5. In the "Y-source" field, define the signal source that supplies the y values of the XY-diagram.
6. To switch the x- and y-values quickly, tap the "Swap XY" button.
7. To maintain a constant ratio while the x- and y-axes are adapted to the acquired data dynamically, activate the "Constant XY-ratio" option.

If the XY-diagram is active or minimized, touch and hold the signal icon to open the "XY-diagram" tab.

8 Measurements

Using the R&S RTO6 you can perform and display different measurements simultaneously, based on the active signal or math waveforms. The color of the results in the result table corresponds with the source waveform color.

The following measurement methods are available:

- Cursor measurements: measurements can be configured for up to 4 cursor sets to determine specific results at the manually defined cursor positions of an active waveform; the results are displayed in a result box.
- Automatic measurements: up to eight measurements can be configured and performed simultaneously on different sources; the results of each measurement are displayed in a result box.
- Quick measurements: performs a set of automatic measurements on the selected waveform at the push of a button. You can configure the set of measurements.

8.1 Cursor measurements

- Cursors and results of cursor measurements..... 284
- Using cursors..... 285
- Settings for cursor measurements..... 288

8.1.1 Cursors and results of cursor measurements

Cursor measurements determine the results at the current cursor positions. The cursors can be positioned manually, or can be configured to follow the waveform. You can measure on one waveform, or on two different waveforms (sources).

Up to 4 cursor sets can be configured and displayed. Each cursor set consists of a pair of horizontal or vertical cursors, or both. Cursor lines can be coupled so that the initially defined distance is always maintained.

The cursors are displayed in the diagrams of the source waveform only, or in all diagrams. For each measurement, labels can be defined for the cursors. By default, the cursors are labeled as Cu1.1, Cu1.2, Cu2.1, Cu2.2, Cu3.1, Cu3.2, Cu4.1, Cu4.2.

How to set up cursor measurements is described in Chapter 8.1.2, "Using cursors", on page 285. The Chapter 8.1.3, "Settings for cursor measurements", on page 288 provides a detailed description of all settings.

Cursors can also define a gate to limit the measurement to the section of the waveform between the cursor lines. See Chapter 8.2.3.2, "Gate settings for measurements", on page 305.

The result display of cursor measurements is configurable. Results can be shown in a docked table, or in floating result boxes for each measurement. Similar to waveform diagrams, you can also minimize the result box to a result icon on the signal bar.

For details on using the result box, see Chapter 3.3.7, "Displaying results", on page 58 and "Result position" on page 85.

8.1.1.1 Cursor measurements on time-based waveforms

The cursor for measurement on time-based waveforms returns the following results. The results are displayed automatically when a cursor measurement is enabled.

Figure 8-1: Measurement results of a cursor measurement in time domain

Label	Description
X1, X2	Time at the position of the vertical cursors.
Y1, Y2	Vertical values of the waveform at the position of the horizontal cursors in V or A.
ΔX	Difference between the vertical cursor (time) values
ΔY	Difference between the horizontal cursor values
1/ΔX	Inverse time difference
ΔY/ΔX	Slope of the waveform between the cursors (if measured on one source)

8.1.1.2 Cursor measurements on spectrum waveforms

If the measurement source is a spectrum waveform, the results have a different meaning. Measurement on 2 spectrum waveforms is not possible.

The result box for measurement on spectrum waveforms shows the following information.

Label	Description
X1, X2	Frequency at the position of the vertical cursors
Y1, Y2	Vertical values of the waveform at the position of the horizontal cursors in dB.
Bw	Difference between the vertical cursor (frequency) values
ΔY	Difference between the horizontal cursor values

To set the cursor lines to the peaks, various functions are available in the "Peak Search" tab, see Chapter 8.1.3.3, "Peak Search tab", on page 293.

8.1.2 Using cursors

You can start cursor measurements by using the "Cursor" icon on the toolbar, or using the [Cursor] key. For detailed configuration, use the "Cursor" dialog box.

- Starting a simple cursor measurement 286
- Configuring a cursor measurement 286
- Configuring the cursor display 288
- Saving measurement results to file 288

8.1.2.1 Starting a simple cursor measurement

To display cursors using the toolbar

1. Tap the "Cursor" icon on the toolbar.

2. Tap the waveform that you want to measure. Alternatively, you can draw a rectangle in the diagram to position the cursor lines.

   The cursor lines appear and the cursor results are displayed in a table or result box.

To display cursors using the [Cursor] key

1. Select the waveform that you want to measure.
2. Press the [Cursor] key.

   The cursor lines and the measurement results are displayed.

To disable one cursor measurement

1. Tap the "Delete" icon on the toolbar.
2. Tap a cursor line.

   The cursor set and its measurement results are deleted.

To disable all cursor measurements

1. Tap the "Delete" icon on the toolbar.
2. Tap the results table.

   All cursor sets and measurement results are deleted.

8.1.2.2 Configuring a cursor measurement

To modify the position of the cursor lines, you can drag the lines on the screen. In addition, various settings are possible to refine the measurement.

The complete configuration of cursor measurements is provided in the "Cursors" dialog.

1. To open the "Cursors" dialog, use one of these ways:
   - Open the "Menu", and select "Cursor".

- Press the [Cursor] key.
- Use the overlay menu:
  - Double-tap a measurement result in the results table.
  - Adjust source, type, and positions of the cursor lines in the overlay menu.
  - Tap "Advanced Setup" to open the "Cursors" dialog.
- If the results are shown in a floating result box, tap the ⚙ icon in the result box.

2. Select the "Setup" tab.
3. Select the subtab for the cursor set that you want to use.
4. Select the source - the measured waveform. You can select any input channel, or various other active waveforms. Available sources are shown in the source list.
5. If necessary, enable and select a 2nd waveform.
6. Select the cursor type: horizontal, vertical, or both.
7. Define the position of the cursors:
   a) To define exact positions of the cursor lines, enter the X-position for each vertical cursor and the Y-position for each horizontal cursor. If it is not possible to set horizontal cursors, disable "Track waveform".
   b) To position the horizontal cursors automatically, select "Track waveform".
   In this case, cursor 1 indicates the current maximum, cursor 2 indicates the current minimum. If both horizontal and vertical cursors are displayed, the horizontal cursors are placed at the crossing points of the vertical cursors with the waveform. Adjust the vertical cursors manually, and the horizontal cursors follow.
   c) If the waveform arithmetics are set to "Envelope", and "Track waveform" is active, select which horizontal cursor is positioned to the maximum and which to the minimum envelope values.
   d) To keep the distance between the vertical cursors when one cursor is moved, select "Coupling (2 follows 1)".
8. Optionally, select "Show in all diagrams" in the "Advanced" tab. This setting enables the cursor display in all diagrams that are in the same domain as the selected source (time or spectrum).
9. To set the cursors for a spectrum measurement to peak values, select the "Peak Search" tab.
   Tap one of the buttons to place the cursors on the selected peak value. For details, see Chapter 8.1.3.3, "Peak Search tab", on page 293.
10. Tap the "Cursor On/Off" button in the "Setup" tab to activate the cursor measurement.

    The cursors lines and the results are displayed. For details on cursor measurement results, see Chapter 8.1.1, "Cursors and results of cursor measurements", on page 284.

8.1.2.3 Configuring the cursor display

By default, the cursors are displayed as lines in the diagrams and labeled according to the syntax: Cu<cursor set number>.<1|2>

For example, the cursors for the cursor set 2 are labeled 2.1 and 2.2. The horizontal and the vertical cursors lines have the same labels.

You can change the default cursor display and labels.

To set the cursor style

1. Open the "Menu" and select "Settings".
2. Select "Appearance".
3. Select the "Cursor" tab.
4. Select the cursor style. See also: "Cursor style" on page 85.

To add labels to cursor lines

1. Open the "Cursors" dialog box, for example, with "Menu" > "Cursor".
2. Select the subtab for the cursor set you want to configure.
3. Select the "Advanced" tab.
4. For each vertical and horizontal cursor, enter a label.
5. Select "Show labels".

8.1.2.4 Saving measurement results to file

1. Open the "Menu" and select "Save/Recall".
2. Select the "Save" tab.
3. Select "Results".
4. Select the results that you want to export.
5. Save the file.

8.1.3 Settings for cursor measurements

Cursor measurements are configured in the "Cursors" dialog box.

8.1.3.1 Setup tab

The "Setup" tab contains the settings for cursor measurements.

The cursor style and position of the measurement results is defined in the "Settings" > "Appeareance" > "Cursor" dialog box.

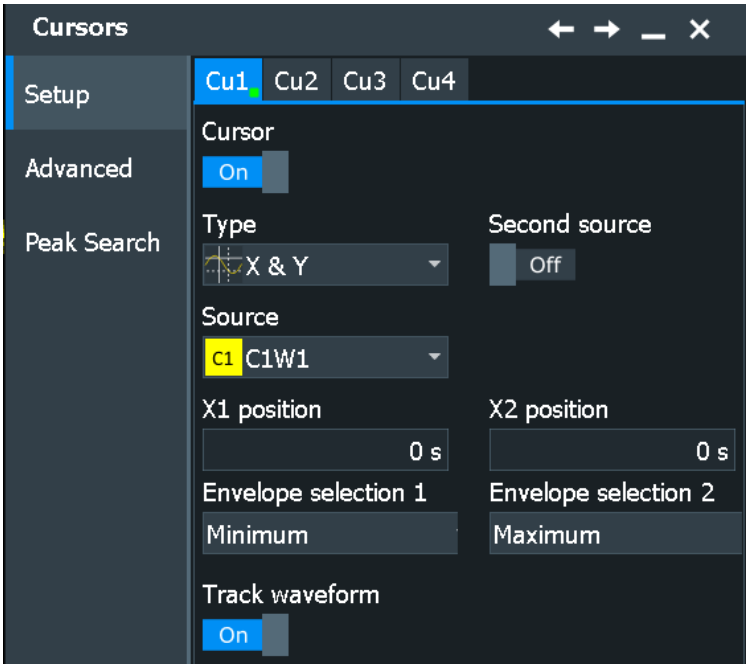


Figure 8-2: Setup for cursor on 1 source, with waveform tracking

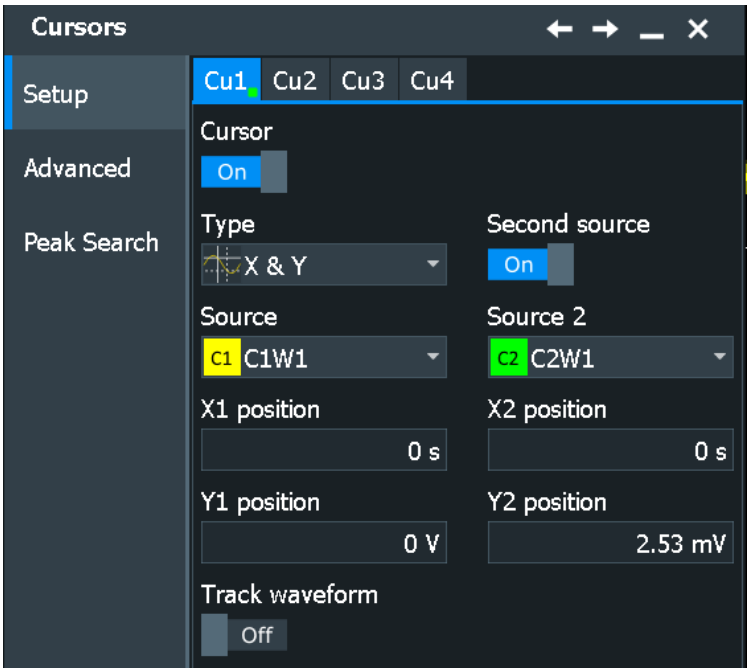


Figure 8-3: Setup for cursor on 2 sources

Cu1, Cu2, Cu3, Cu4

The settings for each cursor measurement (or cursor set) are configured on separate tabs. For each cursor set, a horizontal pair of cursors, a vertical pair of cursors, or both can be displayed.

Cursor On/Off
Enables the selected cursor measurement.

Remote command:
`CURSor<m>:STATe` on page 1469

Type
Defines the cursor type to be used for the measurement.

- Horizontal cursors (Y-cursors)
  Horizontal cursors are positioned automatically along the waveform and can be adjusted manually
- Vertical cursors (X-cursors)
  Both vertical cursor lines are set automatically to the trigger position, and you can reposition them manually.
- Both vertical and horizontal cursors (X&Y-cursors)
  Horizontal cursors are positioned automatically along the waveform and vertical cursors are set to the trigger position. You can reposition all cursor lines manually.

Remote command:
`CURSor<m>:FUNCtion` on page 1469

Source
Defines the source of the cursor measurement. Any of the input signal, math, reference or XY waveforms can be selected.

Remote command:
`CURSor<m>:SOURce` on page 1470

Second source, Source 2
Enables and selects a second source for the cursor measurements. If enabled, the second cursor lines Cx.2 measure on the second source. Using a second source, you can measure differences between two channels with cursors.

Remote command:
`CURSor<m>:USSource` on page 1471
`CURSor<m>:SSOurce` on page 1470

X1 position, X2 position
Defines the position of the vertical cursors.

Remote command:
`CURSor<m>:X1Position` on page 1472
`CURSor<m>:X2Position` on page 1472

Y1 position, Y2 position
Defines the position of the horizontal cursor lines. The setting corresponds to the V1 and V2 values in the "Cursor Results" box.

If "Track waveform" is enabled, the user setting is disabled and the measurement results are displayed in the "Cursor Results" box.

Remote command:
`CURSor<m>:Y1Position` on page 1473
`CURSor<m>:Y2Position` on page 1473

Envelope selection 1|2

Envelope selection is effective under the following conditions:

- The waveform arithmetic of the cursor source waveform is set to envelope waveform (see "Arithmetic" on page 130)
- "Track waveform" is enabled.
- Both horizontal and vertical cursors are enabled ("Type" = *Both*).

The setting defines which horizontal cursor is positioned to the maximum and which to the minimum envelope values.

"Minimum" The horizontal cursor is set to the crossing point of the vertical cursor with the minimum waveform envelope.

"Maximum" The horizontal cursor is set to the crossing point of the vertical cursor with the maximum waveform envelope.

Remote command:
`CURSor<m>:X1ENvelope` on page 1474
`CURSor<m>:X2ENvelope` on page 1475

Track waveform

The horizontal cursors track the waveform. The first cursor line indicates the current vertical minimum, and the second cursor line indicates the maximum. If the waveform changes, e.g. during a running measurement, the cursors move along with it. If both horizontal and vertical cursors are displayed, the horizontal cursors are positioned to the crossing points of the vertical cursors with the waveform. The measurement results are displayed in the "Cursor Results" box.

Tracking disables the Y-coupling (coupling of horizontal cursor lines) and the Y user position settings.

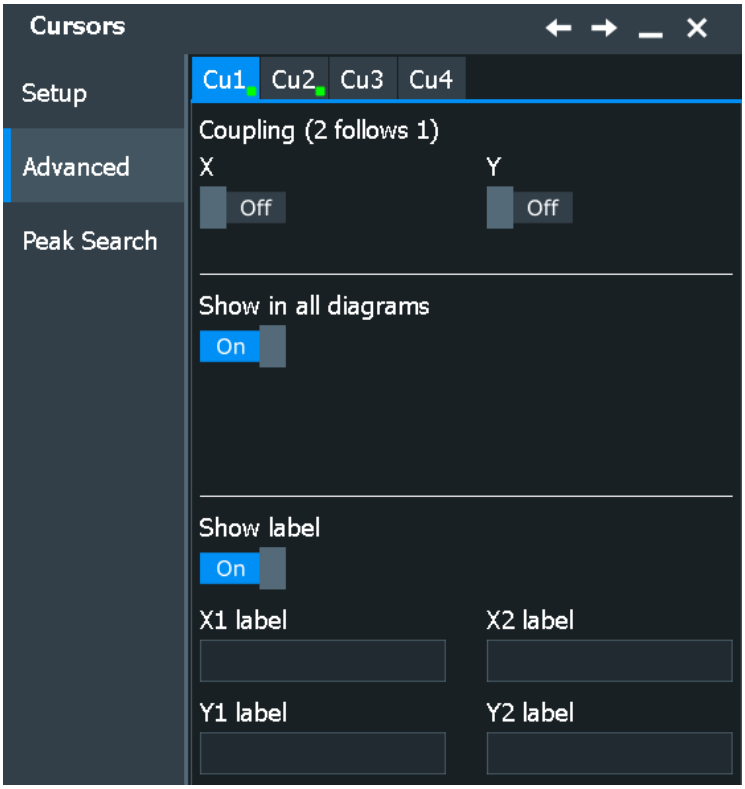
Remote command:
`CURSor<m>:TRACking[:STATe]` on page 1472

8.1.3.2 Advanced settings

Access: [Cursor] key > "Advanced" tab

The settings in the "Advanced" cursor tab configure the behavior and display of cursor lines, and labels for the lines.

The cursor style and position of the measurement results is defined in the "Settings" > "Appeareance" > "Cursor" dialog box. See Chapter 4.3.7, "Cursor appearance settings", on page 85.



Cu1, Cu2, Cu3, Cu4
The settings for each cursor measurement (or cursor set) are configured on separate tabs. For each cursor set, a horizontal pair of cursors, a vertical pair of cursors, or both can be displayed.

Coupling (2 follows 1)
Couple the horizontal and vertical cursor lines so that the distance between the two lines remains the same if one cursor is moved.

Remote command:
`CURSor<m>:YCOupling` on page 1474
`CURSor<m>:XCOupling` on page 1473

Show in all diagrams
Shows the enabled cursor measurements in all active diagrams of the same (time/spectrum) domain.

Remote command:
`CURSor<m>:SIAD` on page 1476

Show label
Shows the cursor labels in the diagram.

Remote command:
`CURSor<m>:LABel[:LABel]` on page 1475

X1 label, X2 label
Defines a label to be displayed with the vertical cursors.

By default, the cursors are labeled as Cu1.X1, Cu1.X2, Cu2.X1, Cu2.X2, Cu3.X1, Cu3.X2, Cu4.X1, Cu4.X2.

Remote command:
`CURSor<m>:LABel:X<1..2>` on page 1475

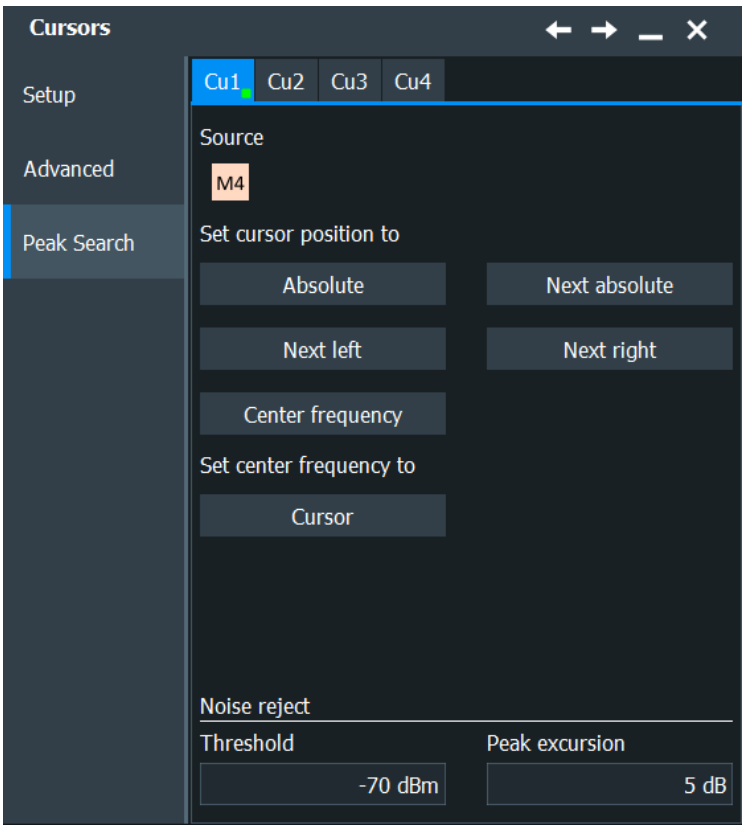
Y1 label, Y2 label
Defines a label to be displayed with the horizontal cursors. By defaYult, the cursors are labeled as Cu1.Y1, Cu1.Y2, Cu2.Y1, Cu2.Y2, Cu3.Y1, Cu3.Y2, Cu4.Y1, Cu4.Y2.

Remote command:
`CURSor<m>:LABel:Y<1..2>` on page 1476

8.1.3.3 Peak Search tab

Access: [Cursor] key > "Peak Search" tab

The settings on this tab are only available in spectrum mode, i.e. the source of the cursor measurement is an FFT math waveform. In this case, the cursors can indicate the results of a peak search on the waveform. You can define which peaks the instrument determines by defining the noise reject settings.



Cu1, Cu2, Cu3, Cu4
The settings for each cursor measurement (or cursor set) are configured on separate tabs. For each cursor set, a horizontal pair of cursors, a vertical pair of cursors, or both can be displayed.

Source
Defines the source of the cursor measurement. Any of the input signal, math, reference or XY waveforms can be selected.

Remote command:
`CURSor<m>:SOURce` on page 1470

Absolute
Both cursors are set to the absolute peak value.

Remote command:
`CURSor<m>:MAXimum[:PEAK]` on page 1478

Next absolute
Cursor 2 is set to the next smaller absolute peak from the current position.

Remote command:
`CURSor<m>:MAXimum:NEXT` on page 1479

Next left
Cursor 2 is set to the next peak to the left of the current position.

Remote command:
`CURSor<m>:MAXimum:LEFT` on page 1478

Next right
Cursor 2 is set to the next peak to the right of the current position.

Remote command:
`CURSor<m>:MAXimum:RIGHt` on page 1478

Center frequency
Sets the vertical cursor line Cu1 to the center frequency.

Remote command:
`CURSor<m>:FFT:TOCenter` on page 1478

Set center frequency to Cursor
Sets the center frequency to the frequency value that is measured at cursor line Cu1.

Remote command:
`CURSor<m>:FFT:SETCenter` on page 1478

Threshold
Defines an absolute threshold as an additional condition for the peak search. Only peaks that exceed the threshold are detected.

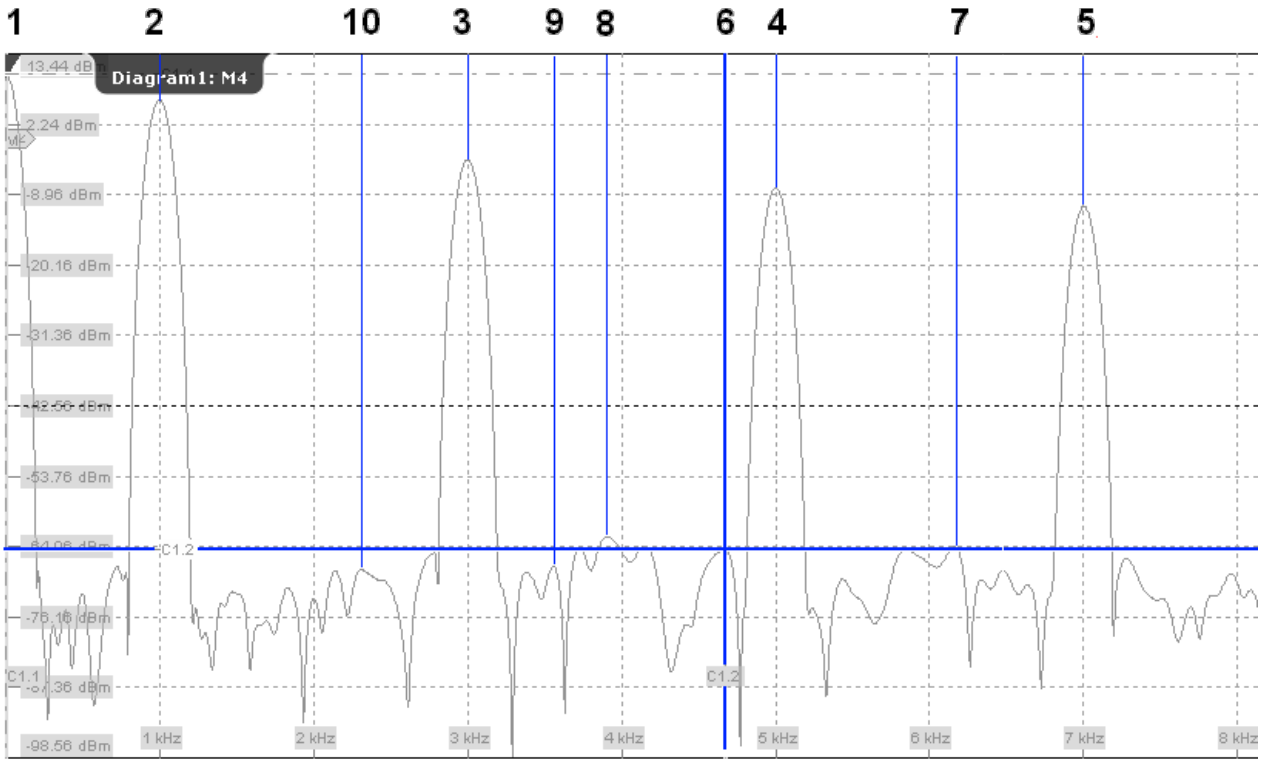
This setting is only available for spectrum waveforms. It is valid for cursor measurements, spectrum measurements and peak search.

Remote command:
`CURSor<m>:THReshold` on page 1631
`MEASurement<m>:SPECtrum:ATHReshold` on page 1514

Peak excursion
Defines a relative threshold, the minimum level value by which the waveform must rise or fall to be considered as a peak. To avoid identifying noise peaks, enter a peak excursion value that is higher than the noise levels.

This setting is only available for spectrum waveforms. It is valid for cursor measurements, spectrum measurements and peak search.

The following figure shows a cursor measurement on a spectrum waveform:



If "Peak excursion" is 30 dB, the peaks 1 to 5 are found. If "Peak excursion" is 20 dB, also the peaks 6 to 10 are found. The cursor position is on peak 6.

Remote command:
`CURSor<m>:PEXCursion` on page 1630
`MEASurement<m>:SPECtrum:PEXCursion` on page 1514

8.2 Automatic measurements

The R&S RTO6 can perform many measurements in parallel. The measurements are combined in up to 8 measurement groups. In addition, you can run the quick measurement.

The basic measurement settings are source, category, and the selection of the measurement. You can refine the setup to get more specific results:

- **Gating**
  A gate limits the measurement to a user-defined part of the waveform. See Chapter 8.2.3, "Measurement gates", on page 304.
- **Statistics and long term measurements**
  To evaluate time-dependent behavior of measurement results, you can use statistics, long term measurements, and tracks. You can also decide, how many measurement results per acquisition contribute to the calculation. See: Chapter 8.2.12, "Result analysis", on page 355.
- **Limit checks and actions on test result**
  Limit and margin checks evaluate if the measurement result exceeds a specified value. You can define actions that are performed on limit or margin violation. See: Chapter 8.2.13, "Limit and margin checks", on page 363.

Measurement categories

The various measurement are grouped in several categories. The category defines which sources can be analyzed.

Time domain:

- Amp/Time: amplitude and time measurements
- Eye: eye diagram measurements
- Histogram: measurements on histograms
- Jitter measurements (available if one of the jitter options is installed)
- Protocol: available for audio signals (option R&S RTO6-K510)
  Protocol: advanced analysis (available if option R&S RTO6-K500 is installed)

Frequency domain

- Spectrum: measurements on spectrum waveforms
- Histogram: measurements on histograms

Details on automatic measurements are described in the following chapters:

- Measurement setup in general....297
- Measurement results....301
- Measurement gates....304
- Reference levels....307
- Amplitude/time measurements....316
- Eye diagram measurements....330
- Spectrum measurements....334
- Histograms and histogram measurements....340
- Jitter measurements....347
- Optical measurements....348
- Protocol measurements (option R&S RTO6-K500)....349
- Result analysis....355
- Limit and margin checks....363

8.2.1 Measurement setup in general

Up to 8 measurement groups can be defined. Each measurement group is configured in its own subtab. For each measurement group, the category and the source are defined in the "Setup" tab, and you can also enable statistic evaluation. Available sources and measurements depend on the selected category, for example, histogram measurements need a defined histogram as source. For each measurement group, you select all required measurements. If further settings are available for a measurement, a settings icon is shown beside the measurement's name.

8.2.1.1 Starting an automatic measurement

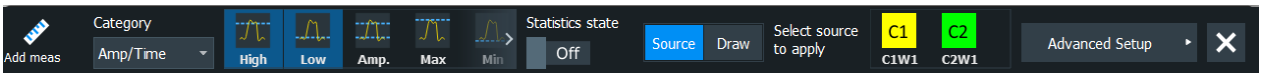
There are three methods to start an automatic measurement, each with slightly different effects:

- Using the "Measure" icon on the toolbar:
  See: "To start a measurement using the toolbar icon" on page 297.
- Using the "Measure" menu.
  Pressing the [Measure] key on the front panel, or using the "Measure" menu.
  See: Chapter 8.2.1.2, "Configuring measurements", on page 297.

To start a measurement using the toolbar icon

1. Tap the "Measurement" icon on the toolbar.

2. On the toolbar assist, select the category, the measurements and the source to be measured.



3. Define the measurement range in one of these ways:
   - To measure the complete waveform, tap the diagram with the waveform.
   - To define a gate that limits the measurement, draw a rectangle on the screen.

   The "Meas Results" are displayed.

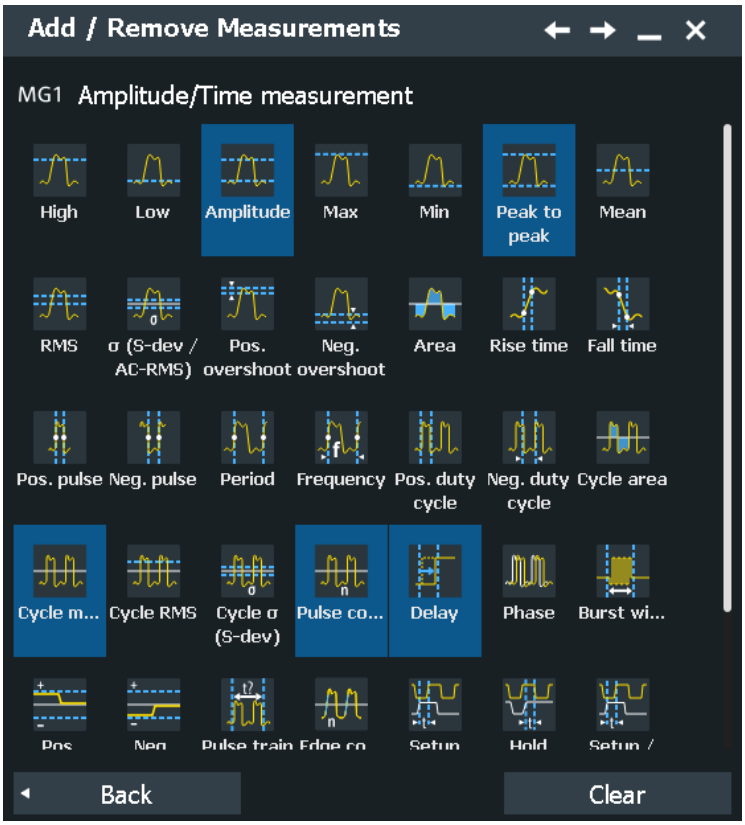
8.2.1.2 Configuring measurements

1. Press the [Measure] key on the front panel, or open the menu and select "Measure".

   The "Measurements" dialog box opens.
   If you double-tap the title if a result table, the toolbar assist opens. Here you can change the measurement types and enable statistics. If you tap "Advanced Setup", the "Measurements" dialog box opens.

2. Select the subtab for the measurement group that you want to configure.

3. Select the measurement "Category", for example, "Amp/Time".
4. Tap "Source" and select the waveform to be measured.
   Spectrum measurements require an FFT math waveform as measurement source.
   Histogram measurements require a histogram as source.
5. Tap "Add / Remove" and select all measurements that you want to include in the measurement group. Tap "Clear" to deselect all settings.



6. Tap "Back" to close the selection box.
   All selected measurements are displayed in the list of active measurements.
7. A "Settings" icon indicates whether further settings are required. Tap the measurement. A dialog box opens, and you can adjust the measurement.

   The settings are explained in the following chapters:
   - Chapter 8.2.5.2, "Settings for amplitude/time measurements", on page 321
   - Chapter 8.2.6.2, "Settings for eye diagram measurements", on page 333
   - Chapter 8.2.7.2, "Settings for spectrum measurements", on page 336
   - Chapter 8.2.8.5, "Settings for histogram measurement", on page 346
8. Optionally, define a gate to restrict the measurement to a part of the waveform. See Chapter 8.2.3.1, "Using measurement gates", on page 304.

   If you enabled the measurement with the toolbar icon and drew a rectangle on the diagram, the gate is already defined and enabled.

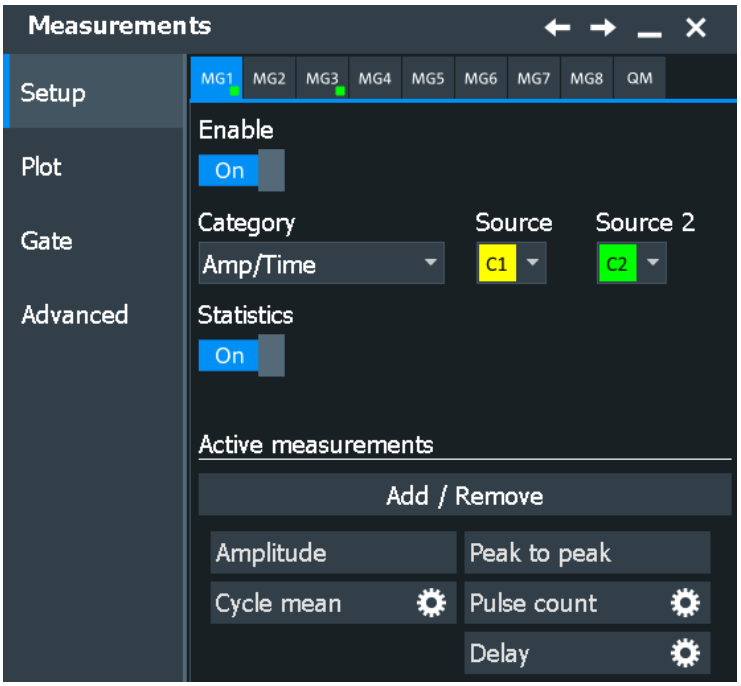
9. To compile and display statistics for the measurement, enable "Statistics". See also Chapter 8.2.12.1, "Statistics", on page 356.
10. Optionally, perform a limit or margin check as described in Chapter 8.2.13.1, "Performing limit checks", on page 364.
11. "Enable" the measurement group.

    The measurement results are displayed by default below the waveform diagram.

8.2.1.3 General measurement settings

Automatic measurements are configured in the "Measurements" dialog box.

Access: [Measure] > "Setup" tab



This section describes the settings that relate to all measurements. Specific settings are described in the corresponding category chapters:

- Chapter 8.2.5.2, "Settings for amplitude/time measurements", on page 321
- Chapter 18.1, "Jitter measurements (options R&S RTO6-K12/K13)", on page 1069
- Chapter 8.2.6.2, "Settings for eye diagram measurements", on page 333
- Chapter 8.2.7.2, "Settings for spectrum measurements", on page 336
- Chapter 8.2.8.5, "Settings for histogram measurement", on page 346

MG 1/2/3/4/5/6/7/8/Quick Meas
Subtabs for each measurement group, and for the quick measurement. The subtabs contain the configuration settings. A green dot on the tab indicates that the measurement group is active.

Enable
Starts the measurements of the selected measurement group.

Remote command:
`MEASurement<m>[:ENABle]` on page 1480

Category
Measurement category. The following categories are available:

Time domain:
- Amp/Time: amplitude and time measurements
- Eye: eye diagram measurements
- Histogram: measurements on histograms
- Jitter measurements (available if one of the jitter options is installed)
- Protocol: available for audio signals (option R&S RTO6-K510)
  Protocol: advanced analysis (available if option R&S RTO6-K500 is installed)

Frequency domain
- Spectrum: measurements on spectrum waveforms
- Histogram: measurements on histograms

Remote command:
`MEASurement<m>:CATegory` on page 1482

Source, Source 2
Define the source of the measurement group. The 2nd source is required for amplitude/time measurements that are performed on two waveforms (e.g. delay, phase). Availability of sources depends on the selected category and installed options.

Remote command:
`MEASurement<m>:SOURce` on page 1480

Active measurements, Add / Remove
The "Active Measurements" list shows the measurements that are selected for the current category. At least, one measurement must be selected. If further settings are available for a measurement, a settings icon is shown beside the measurement's name.

To change the selection, tap "Add / Remove Measurements".

In the "Add / Remove Measurements" dialog box, select the measurements that you need. Tap "Clear" to disable all selected measurements. Tap "OK" to confirm the selection.

Remote command:
`MEASurement<m>:MAIN` on page 1483
`MEASurement<m>:ADDitional` on page 1484

Envelope
This setting is only available for measurements on envelope waveforms, see "Arithmetic" on page 130.

"Both"	The upper and the lower envelope are used in measurements. For time measurements, the averages of min and max values are used, that is, the measurement is performed on the average waveform built from the upper and lower envelope.
"Maximum"	Measurements are performed on the upper envelope.

"Minimum" Measurements are performed on the lower envelope.

Remote command:
`MEASurement<m>:ENVSelect` on page 1492

Statistics
Enables the calculation and display of statistical results.

Remote command:
`MEASurement<m>:STATistics[:ENABle]` on page 1533

8.2.2 Measurement results

By default, the results of automatic measurements are displayed below the waveform diagram when an automatic measurement is enabled.

Meas Group 1		Meas Group 2	
High	122.53 mV	Rise time	11.847 ns
Low	-124.51 mV	Fall time	---
Amplitude	247.04 mV	Pos. pulse	20.05 ns
Frequency	25 MHz	Neg. pulse	20 ns
Cycle RMS	87.999 mV		

If you want to save space in the display, drag the results to the signal bar. The most important results are displayed and updated in a results icon.

The function "Clear all" in the "Display" menu resets all results including long-term measurement and statistic results, and also deletes all waveforms and the history.

Which results are displayed depends on the selected measurements and is described in detail in the following chapters.

The following additional results are available:

- **Statistics**
  You can enable statistical evaluation of the measurement results, and select the statistical results that you want to see. Statistic information is provided in the result box. Stopping and restarting the acquisition does not reset statistics but only stops and continues them.
  See Chapter 8.2.12, "Result analysis", on page 355
- **Measurement histograms**
  The results of measurements can be displayed in a histogram which shows the density distribution of the measurement results in a graphic and thus illustrates the statistics of the measurements.
  See Chapter 8.2.8, "Histograms and histogram measurements", on page 340
- **Long-term measurements**
  Long-term measurements show the behavior of measurement results over a longer time or for many samples. You can define the number of long-term points and export the long-term data, including statistical results. The measurement histogram is a vertical histogram shown in the long-term diagram.

Long-term measurements are performed on the measurement that is selected for analysis and math on the "Result Analysis" tab.
See: Chapter 8.2.12, "Result analysis", on page 355

- **Intermediate results**
  You can display auxiliary result lines and reference levels in the source diagram, see Chapter 8.2.2.2, "Configuring the results display", on page 303.

Remote commands:

- `MEASurement<m>:ARES?` on page 1486
- `MEASurement<m>:ARNames` on page 1486
- `MEASurement<m>:RESult[:ACTual]?` on page 1487
- `MEASurement<m>:RESult:COUNt?` on page 1489

8.2.2.1 Measurement status

The overall status of measurement results is indicated by various icons. In general, a question mark before the result value indicates that the measurement result might not be correct due to insufficient amplitude level. Check your amplitude and reference level settings. The icon colors indicate the state of the limit and margin checks.

Icon	Description
No icon, no result value ("---")	The instrument cannot measure the required value, for example, if the acquisition does not contain at least one complete period for frequency and cycle measurements. Check and adjust the waveform settings to get results.
	The measurement result might not be correct due to insufficient amplitude level. Check your amplitude and reference level settings. Limit and margin checks are disabled.
	Limit and margin checks passed, measurement results are reliable.
	The measurement result might not be correct due to insufficient amplitude level. Check your amplitude and reference level settings. Limit and margin checks passed.
	The measurement result might not be correct due to insufficient amplitude level. Check your amplitude and reference level settings. Margin checks failed.
	Margin checks failed.
	The measurement result might not be correct due to insufficient amplitude level. Check your amplitude and reference level settings. Limit checks failed.
	Limit checks failed.

8.2.2.2 Configuring the results display

The measurement results can be displayed in a table below the waveform diagrams, in a floating result box, or in a minimized result icon on the signal bar.

The display settings for measurements are provided on the "Gate/Display" tab, see Chapter 8.2.2.3, "Display settings for results", on page 303.

To clear the measurement results

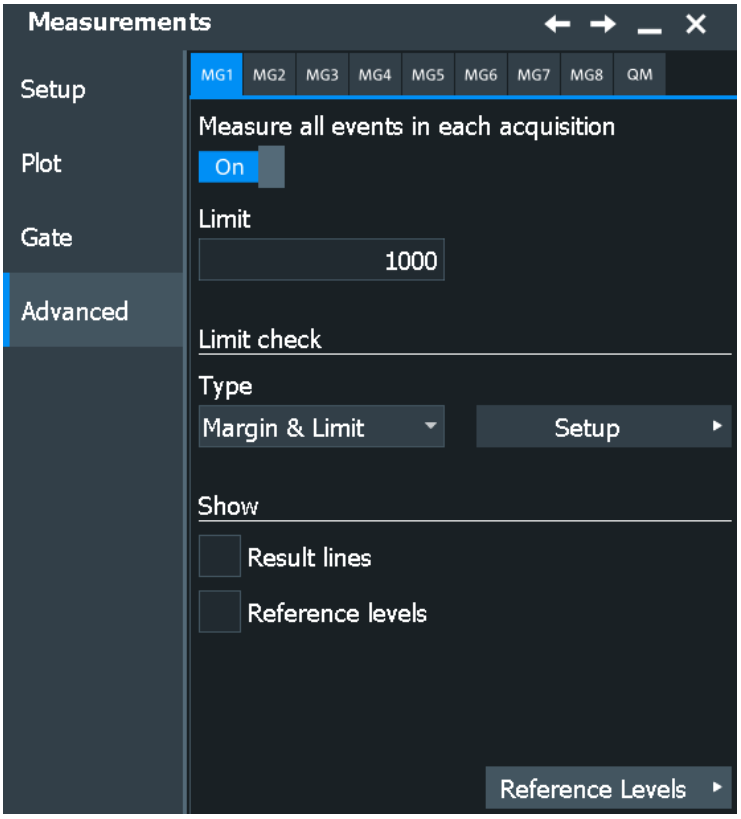
1. On the "Display" menu, tap "Clear all".
2. To restart measurement statistics, without deleting other results, select "Reset" on the "Result Analysis" tab.

   The results in the selected measurement result box are cleared and written anew.

8.2.2.3 Display settings for results

Display settings are specific for each measurement group.

Access: [Measure] > "Advanced" tab



General display setting for measurements are provided in "Settings" > "Appearance" > "Measurements", see Chapter 4.3.8, "Measurement appearance settings", on page 86.

Result lines
Result lines mark the samples in the waveform that are used to obtain the measurement result. For example, maximum, minimum, high and low values, and mean are merked by horizontal lines; start and end are marked by vertical lines.

Remote command:
`MEASurement<m>:DISPlay:RESults` on page 1532

Reference levels
Displays the reference levels used for the measurement in the diagram.

Remote command:
`MEASurement<m>:DISPlay:LEVels` on page 1531

Result control panel
Extends the result box of the selected measurement group with the source settings and the statistics enabling. Thus you can check and change the measurement sources directly in the results box, and also enable statistics there. The setting is only available if the result boxes are not grouped ("Group result dialogs" = Off).

8.2.3 Measurement gates

Gate areas limit the measurement to a user-defined range of the waveform. The gate settings are defined on the "Gate " tab.

Each measurement group can use its own gate. Make sure to select the correct measurement group.

8.2.3.1 Using measurement gates

If you have enabled the measurement with the toolbar icon and drew a rectangle on the diagram, the gate is automatically defined and enabled. If you want to create a gate for a running measurement, or if you want to modify the gate area, configuration is done in the "Measurements" > "Gate" dialog box.

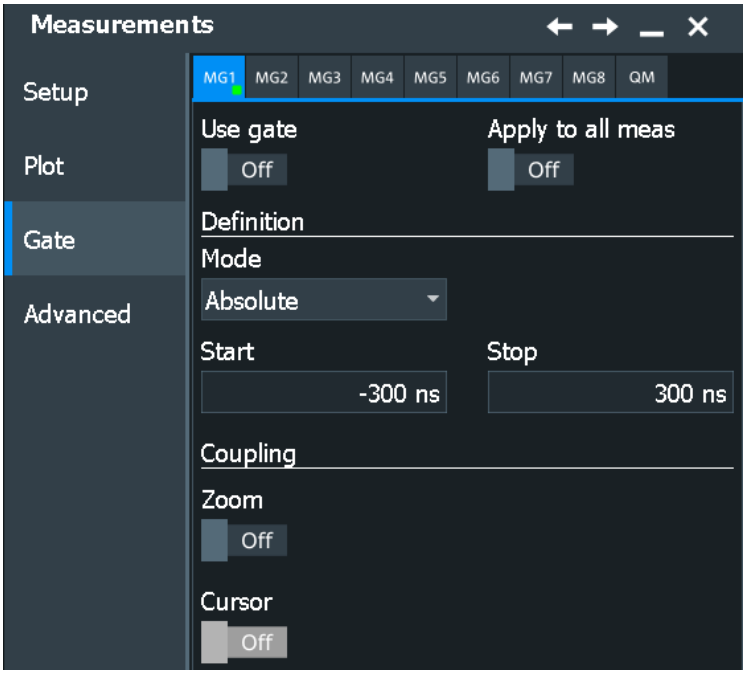
1. In the "Measurements" dialog, select the "Gate" tab.
2. Select the subtab for the measurement group you want to configure.
3. To define the gate, use one of the following methods:
   - Define the start value and the stop value of the gate area by entering either absolute or relative values.
   - If a zoom area has already been defined for the waveform, couple the gate area to the zoom area by selecting the "Zoom coupling" option.
   - If a cursor measurement has already been defined for the waveform, couple the gate area to the cursor lines by selecting the "Cursor coupling" option.
4. If you want to use the same gate for all measurement group, enable "Apply to all meas".

5. Tap the "Use gate" icon to enable the gate usage.

   The measurement is performed on the selected part of the waveform. The gate is shown in the diagram.

8.2.3.2 Gate settings for measurements

Access: [Measure] > "Gate" tab



Result display settings are described in Chapter 8.2.2.3, "Display settings for results", on page 303.

Use gate
Considers the gating settings for the selected measurement and displays the gate.

Remote command:
`MEASurement<m>:GATE[:STATe]` on page 1546

Apply to all meas
If you enable the gate coupling, the gate settings of the selected measurement are copied to all other measurements. Thus, all measurements use the same gate. If zoom or cursor coupling is active in a measurement, the zoom size and cursor positions are adjusted.

Remote command:
`MEASurement<m>:GATE:GCOupling` on page 1549

Gate description
Defines whether the gate settings are configured using absolute or relative values.

"Absolute" The gate is defined by absolute start and stop values.

"Relative" The gate's start and stop values are defined by a percentage of the value range.

Remote command:
CALCulate:MATH<m>:FFT:GATE:MODE on page 1573
MEASurement<m>:GATE:MODE on page 1547
SEARch:GATE:MODE on page 1632

(Relative) Start
Defines the starting value for the gate.

Remote command:
CALCulate:MATH<m>:FFT:GATE:ABSolute:STARt on page 1573
CALCulate:MATH<m>:FFT:GATE:RELative:STARt on page 1574
MEASurement<m>:GATE:ABSolute:STARt on page 1547
MEASurement<m>:GATE:RELative:STARt on page 1547
SEARch:GATE:ABSolute:STARt on page 1632
SEARch:GATE:RELative:STARt on page 1633

(Relative) Stop
Defines the end value for the gate.

Remote command:
CALCulate:MATH<m>:FFT:GATE:ABSolute:STOP on page 1573
CALCulate:MATH<m>:FFT:GATE:RELative:STOP on page 1574
MEASurement<m>:GATE:ABSolute:STOP on page 1547
MEASurement<m>:GATE:RELative:STOP on page 1547
SEARch:GATE:ABSolute:STOP on page 1632
SEARch:GATE:RELative:STOP on page 1633

Zoom
Zoom coupling is available if a zoom is defined. As long as "Zoom coupling" is enabled, the gate area is defined identically to the zoom area - if you change the zoom, the gate changes as well.

If several zoom diagrams are defined, select the zoom diagram to be used for gating. The "Start" and "Stop" values of the gate are adjusted accordingly.

Zoom coupling can be set for measurement gates, FFT gates, and search gates. The zoom must be defined on the diagram that contains the signal source of the measurement, FFT, or search.

Remote command:
MEASurement<m>:GATE:ZCOupling on page 1548
MEASurement<m>:GATE:ZDIagram on page 1549
CALCulate:MATH<m>:FFT:GATE:ZCOupling on page 1574
SEARch:GATE:ZCOupling on page 1633
SEARch:GATE:ZDIagram on page 1634

Cursor
If enabled, the gate area is defined by the cursor lines of an active cursor measurement.

If several cursor measurements are enabled, select the cursor set to be used for gating. The "Start" and "Stop" values of the gate are adjusted to the values of the cursor line positions. The measurement is limited to the part of the waveform between the cursor lines.

Remote command:
`MEASurement<m>:GATE:CCOupling` on page 1548
`MEASurement<m>:GATE:CURSor` on page 1548

8.2.4 Reference levels

Some measurement require reference levels to obtain the measurement points, e.g. time measurements or pulse count. Reference levels are referred to the signals, for each waveform you can define specific reference levels. Thus, for all measurements on a waveform the same reference levels are used.

Usually, reference levels are determined automatically. The instrument determines the high and low signal levels based on amplitude and histogram measurements of the acquisition. The reference levels are set relatively to the determined signal levels.

However, for irregular data and in special measurement setups it may be useful to configure the levels manually:

- Data signals can contain intervals where no data is transmitted, so that a high and low state cannot be determined for each acquisition. In this case, you can define the high and low signal levels manually to evaluate other measurement results.
- If the signal levels vary strongly or have large overshoots, the rise and fall levels may be difficult to determine.
- If fixed levels are defined for the DUT, you can configure the reference levels in the R&S RTO6 correspondingly and analyze the resulting measurement data.

In manual configuration, the reference levels can be set relatively to defined signal levels or as absolute values. You can also set the reference levels directly.

In addition to reference and signal levels, you can define a hysteresis for the middle reference level and tubes for signal levels. Hysteresis is useful for measurements that determine zero-crossings. Period, frequency, and pulse measurements are based on hysteresis - the instrument returns results if the amplitude of the signal exceeds the hysteresis. Thus, measurement during the transient oscillation is also possible. Tubes define evaluation ranges for measurements that require detection of the high level or low level. If the signal value remains within the defined tubes, it is considered to be high or low.

Reference levels and result lines can be displayed in the diagram, see "To display reference levels and result lines" on page 310.

8.2.4.1 Configuring reference levels

To determine reference and signal levels automatically

By default, the histogram of the measurement data is evaluated to determine the required levels automatically. However, you can define several parameters to adapt the evaluation to your data.

1. Open "Menu" > "Measure" > "Advanced" tab > "Reference Level" > "Levels" tab.
2. Define the "Source", the waveform for which the reference is defined. The source can be any signal input, math or reference waveform.
3. Select "Mode" = "Auto" .
4. Define the "Signal level", the method which is used to determine the signal levels. For details, see "Signal level" on page 312.
5. By default, the lower reference level is defined at 10% of the signal amplitude, the middle reference level at 50% and the upper reference level at 90%. You can select other "Relative levels" to be used for evaluation.
   If default percentages do not fit, select "User defined" and enter the percentages for the upper, middle, and lower reference levels.
   The signal levels are determined by the instrument.
6. To determine the reference levels using average values from several histograms, enable the "Histogram averaging" option and define an "Average Count" to define how many histograms are averaged.
   Averaging is not available if "Absolute peak" are selected as the "Signal level".

To determine reference levels manually

You can configure the reference levels manually as fixed absolute or relative values.

1. Open "Menu" > "Measure" > "Advanced" tab > "Reference Level" > "Levels" tab.
2. Define the "Source", the waveform for which the reference levels are defined. The source can be any signal input, math or reference waveform.
3. Select "Mode" = "Manual" .

4. Under "Level definition", select whether you want to define the levels using absolute or relative values.
5. Under "User level", select whether you want to configure the high and low signal levels ("Signal") or the lower, middle and upper reference levels ("Reference").
6. To define high and low signal levels if "Signal" is selected:
   a) Enter the absolute high and low signal levels.
   b) If "Level definition" is relative, select one of the predefined "Relative levels". If default percentages do not fit, select "User defined" and enter the percentages for the upper, middle, and lower reference levels.
   The upper and lower reference levels are computed from the signal level values and the percentage values.
   c) If "Level definition" is absolute, set the absolute "Top distance" and "Bottom distance" values, the differences between signal and reference levels.
   The upper and lower reference levels are computed from the signal level values and the distances.
7. To define lower, middle and upper reference levels if "User reference level" is selected:
   a) Enter the absolute upper and lower reference levels.
   b) If "Level definition" is relative, select one of the predefined "Relative levels". If default percentages do not fit, select "User defined" and enter the percentages for the upper, middle, and lower reference levels.
   The high and low signal levels are computed from the reference level values and the percentage values.
   c) If "Level definition" is absolute, set the absolute "Top distance" and "Bottom distance" values, the differences between signal and reference levels.
   The high and low signal levels are computed from the reference level values and the distances.

To define hysteresis and tubes

1. To define a hysteresis for the middle reference level:
   a) Select the "Advanced" tab.
   b) Enter a percentage of the selected signal level.

   A rise or fall from the middle reference value that does not exceed the hysteresis is rejected and not considered a zero-crossing.
2. To define a tube for the high and low signal levels:
   a) In the "Relative outer (RO)" field, define a percentage of the signal level by which the absolute signal level may be larger than high signal level or lower than the low signal level.
   b) In the "Relative inner (RI)" field, define a percentage of the signal level by which the absolute signal level may be higher than the low signal level or lower than the high signal level.

To display reference levels and result lines

1. Open "Menu" > "Measure" > "Advanced" tab.
2. Select the tab for the measurement you want to configure.
3. Enable "Result lines" or "Reference levels" option, or both.

   The reference levels and intermediate results are displayed in the waveform diagram.

8.2.4.2 Level settings

Access: "Menu" > "Measure" > "Advanced" tab > "Reference Level" > "Levels" tab.

On the "Levels" tab, you define how the reference levels are calculated, or you set them directly.

The following setups are available, according to the selected "Mode":

Automatic reference level mode

In automatic reference level mode, the reference levels are always relative values. You can select one of the predefined sets, or define individual percentage values.

Figure 8-4: Automatic reference level definition

Manual reference level mode

In manual reference level mode, relative and absolute level definitions are possible.

In manual reference level mode with relative level definition, you define the absolute values of high and low signal levels or reference levels, and the reference levels as percentages of the signal amplitude.

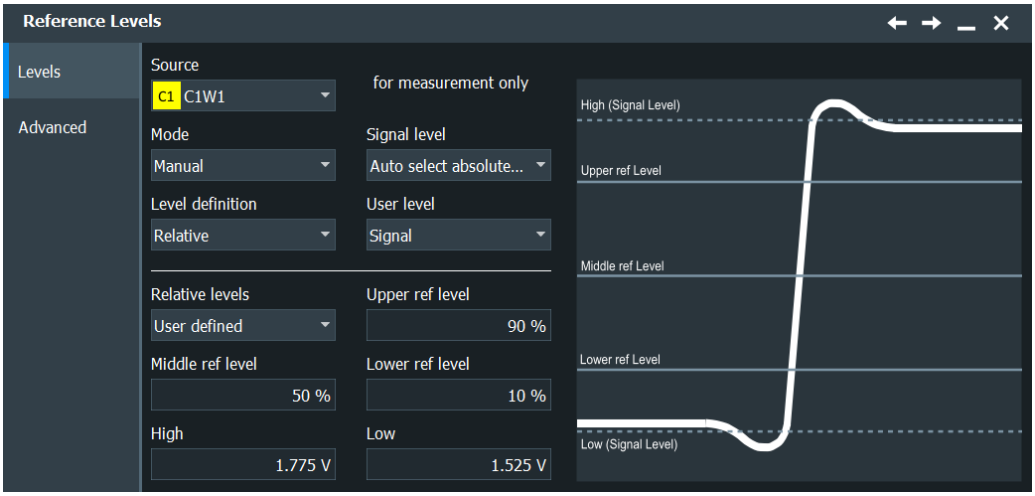


Figure 8-5: Manual reference level mode, relative level definition

In manual reference level mode with absolute level definition, you define the absolute values of high and low signal levels or reference levels, and the distances between reference and signal levels.

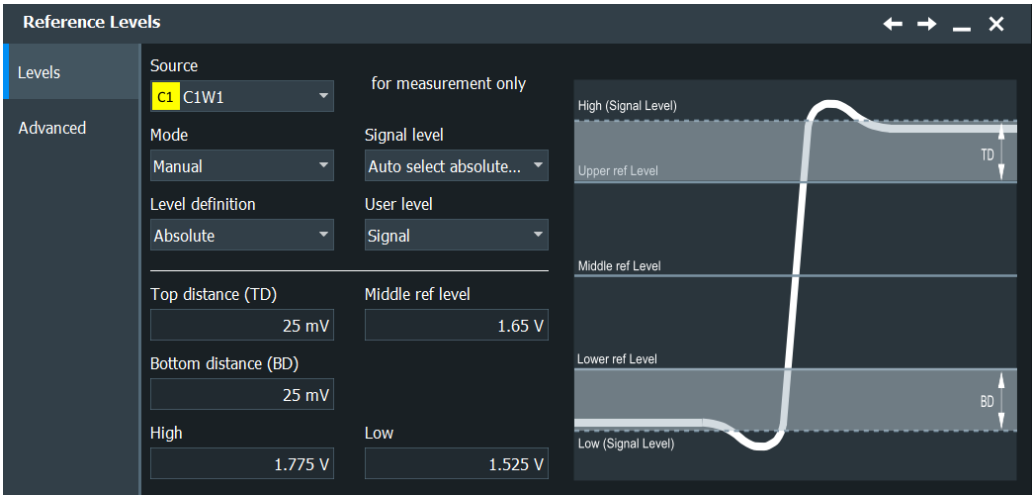


Figure 8-6: Manual reference level mode, absolute level definition

Source

Defines the source for which the reference levels are defined. The source can be any signal input, math or reference waveform.

Remote command:
Source is defined by suffix <m> in "REFLevel" subsystem, see Chapter 24.12.15, "Reference levels", on page 1552

Mode

Defines whether the reference level is configured manually or automatically.

Remote command:
`REFLevel<m>:LDETection` on page 1552

Signal level
Defines the computation method for high and low signal levels. The instrument analyzes the signal, performs amplitude and histogram measurements, and defines the signal levels using the selected method.

The selected method is used to compute the signal levels for determination of reference levels in automatic reference level mode. It is also used for high, low, and amplitude measurements in automatic and manual reference level modes.

See also: Chapter 8.2.8, "Histograms and histogram measurements", on page 340

"Auto select absolute probability"	The most suitable signal levels for the selected measurement are used.
"Peak probability"	The signal levels with the highest probability values are used. These are the upper peak value and the lower peak value of the histogram measurement.
"Mean probability"	The signal levels with mean probabilities are used.
"Absolute peak"	The absolute peak signal levels are used. These are the maximum and minimum signal values of the amplitude measurement.
"Upper absolute peak - Lower mean probability"	The high signal level is the upper absolute peak (the maximum signal level), and the low signal level is the level with the mean probability in the lower half of the histogram.
"Upper mean probability - Lower absolute peak"	The high signal level is the level with mean probability in the upper half of the histogram, and the low signal level is the lower absolute peak (the minimum signal level).
"Upper absolute peak - Lower manual"	The high signal level is the maximum result value of the amplitude measurement. The low signal level is manually set using "Low".
"Upper manual - Lower absolute peak"	The high signal level is set manually using "High". The low signal level is the minimum result value of the amplitude measurement.

Remote command:
`REFLevel<m>:AUTO:MODE` on page 1554

Level definition
In manual reference level mode, the setting defines whether the reference is configured using absolute or relative values.

Remote command:
`REFLevel<m>:LMODe` on page 1553

User level
In manual reference level mode, the setting defines whether the user-defined signal levels or user-defined reference levels are used for the measurements.

"Signal" You can define the high and low signal levels.

"Reference" You can define the reference levels.

Remote command:
`REFLevel<m>:USRLevel` on page 1554

Relative levels
Sets the lower, middle and upper reference levels, defined as percentages of the signal amplitude.

Available relative levels:

- 5/50/95
- 10/50/90
- 20/50/80
- User defined: Enter "Upper ref level", "Middle ref level", and "Lower ref level".

For example, for "5/50/95" the levels are set to the following values:

- Lower reference level = 5% of the signal amplitude
- Middle reference level = 50% of the signal amplitude
- Upper reference level = 95% of the signal amplitude

Remote command:
`REFLevel<m>:RELative:MODE` on page 1553

Upper ref level, Middle ref level, Lower ref level
Define the reference levels in percent, if "Relative levels" is set to "User-defined".

Remote command:
`REFLevel<m>:RELative:UPPer` on page 1559
`REFLevel<m>:RELative:MIDDle` on page 1560
`REFLevel<m>:RELative:LOWer` on page 1560

High
Sets the high signal level.

The high signal level is set in manual reference level mode, for absolute level definition and user signal level selection.

Remote command:
`REFLevel<m>:ABSolute:HIGH` on page 1556

Low
Sets the low signal level.

The low signal level is set in manual reference level mode, for absolute level definition and user signal level selection.

Remote command:
`REFLevel<m>:ABSolute:LOW` on page 1557

Middle
For user signal level selection, the level is the middle level between high and low signal level. The value is adjusted automatically if you change the high or low signal levels. Vice versa, if you change the middle level, the high and low signal levels are adjusted.

For user reference level selection, the level is the middle level between upper and lower reference level. The value is adjusted automatically if you change the upper or lower reference levels. Vice versa, if you change the middle level, the upper and lower reference levels are adjusted.

Remote command:
`REFLevel<m>:ABSolute:MLEVel` on page 1558

Top distance
The distance between the high signal level and the upper reference level - for manual reference level mode and absolute level definition.

Remote command:
`REFLevel<m>:ABSolute:TDIStance` on page 1557

Bottom distance
The distance between the lower reference level and the low signal value - for manual reference level mode and absolute level definition.

Remote command:
`REFLevel<m>:ABSolute:BDIStance` on page 1558

Upper level
The upper reference level required, for example, to determine a rise - for manual reference level mode, absolute level definition and user reference level.

Remote command:
`REFLevel<m>:ABSolute:ULEVel` on page 1558

Lower level
The lower reference level required, for example, to determine a fall - for manual reference level mode, absolute level definition and user reference level.

Remote command:
`REFLevel<m>:ABSolute:LLEVel` on page 1559

Histogram averaging
Enables averaging over several histograms to determine the reference levels.

This function is only available in automatic reference level mode.

Remote command:
`REFLevel<m>:AUTO[:STATe]` on page 1555

Average Count
Defines the number of histograms to calculate the average from.

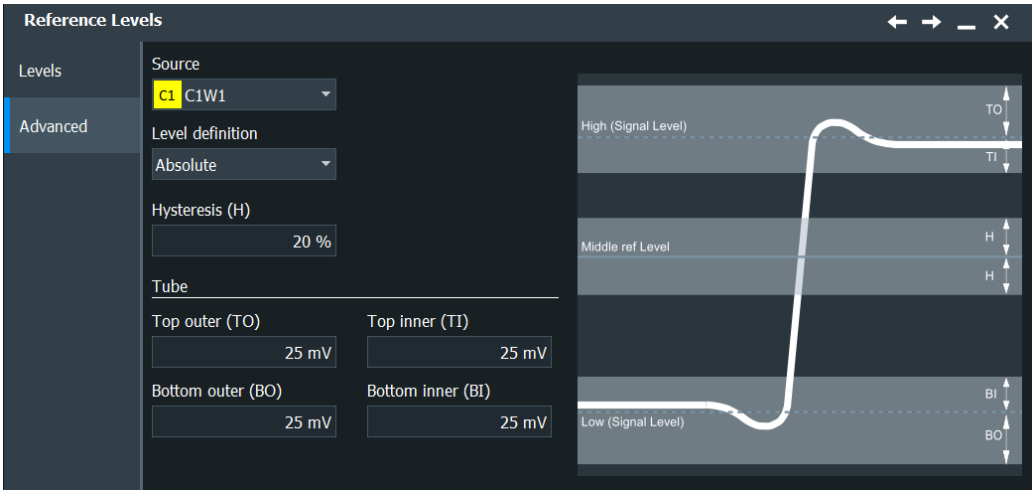
This function is only available in automatic reference level mode.

Remote command:
`REFLevel<m>:AUTO:COUNt` on page 1556

8.2.4.3 Advanced tab

Access: "Menu" > "Measure" > "Advanced" tab > "Reference Level" > "Advanced" tab.

This tab allows you to define evaluation tubes for measurements that require high-level or low-level detection. If the signal value remains within the defined tubes, it is considered to be high or low.



See also: Source and Level definition.

Hysteresis

Defines a hysteresis for the middle reference level. A rise or fall from the middle reference value that does not exceed the hysteresis is rejected as noise.

Remote command:

`REFLevel<m>:RELative:HYSTeresis` on page 1561

Top outer

Defines an area above the high signal level which is still considered to be high level.

Remote command:

`REFLevel<m>:ABSolute:TOTube` on page 1562

`MEASurement<m>:REFLevel:RESult:TOUTer?` on page 1565

Top inner

Defines an area beneath the high signal level which is still considered to be high level.

Remote command:

`REFLevel<m>:ABSolute:TITube` on page 1562

`MEASurement<m>:REFLevel:RESult:TINNer?` on page 1564

Bottom inner

Defines an area above the low signal level which is still considered to be low level.

Remote command:

`REFLevel<m>:ABSolute:BITube` on page 1562

`MEASurement<m>:REFLevel:RESult:BINNer?` on page 1564

Bottom outer

Defines an area beneath the low signal level which is still considered to be low level.

Remote command:

`REFLevel<m>:ABSolute:BOTube` on page 1563

`MEASurement<m>:REFLevel:RESult:BOUTer?` on page 1564

Relative outer
Defines a percentage of the signal level by which the absolute signal level may be larger than the high signal level or lower than the low signal level to be considered high or low, respectively.

Remote command:
`REFLevel<m>:RELative:OTUBe` on page 1561

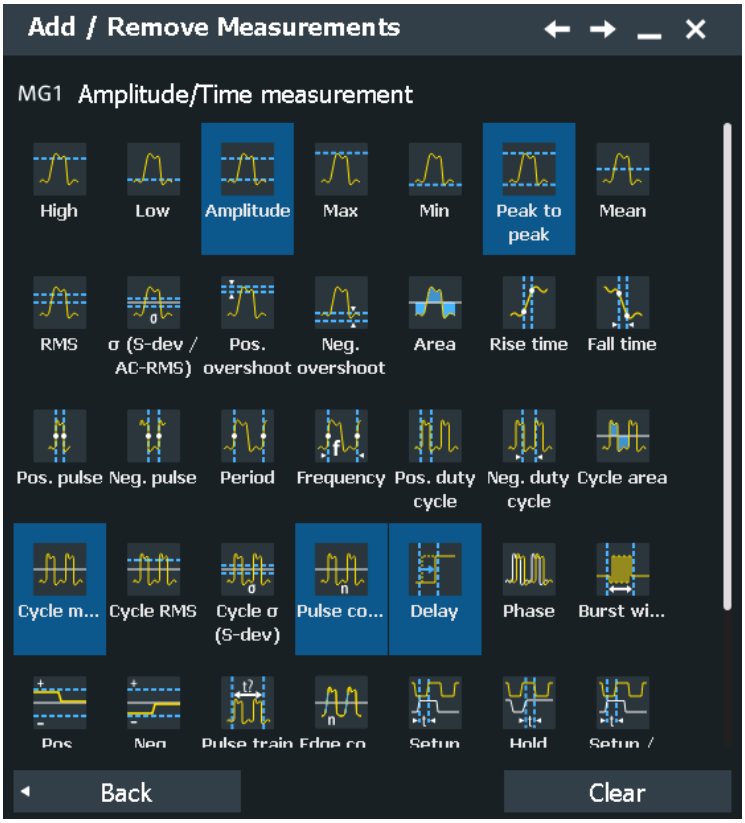
Relative inner
Defines a percentage of the signal level by which the absolute signal level may be higher than the low signal level or lower than the high signal level to be considered low or high, respectively.

Remote command:
`REFLevel<m>:RELative:ITUBe` on page 1561

8.2.5 Amplitude/time measurements

8.2.5.1 Overview of amplitude/time measurements

Access: [Measure] > "Setup" tab > "Amp/Time" category



The R&S RTO6 provides various voltage, time, area and counting measurements in the category "Amp/Time". Some measurements require reference levels to be set according to the measurement purpose.

Reference levels are explained in Chapter 8.2.4, "Reference levels", on page 307.

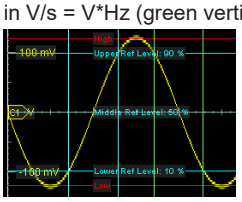
- Amplitude measurements 317
- Time measurements 318
- Area measurements 320
- Counting 321
- Arithmetic 321

Amplitude measurements

Access: [Measure] > "Setup" tab > "Amp/Time" category

Table 8-1: Amplitude measurements

	Meas. type	Symbol	Description/Result
	High	X_{High}	High signal level
	Low	X_{Low}	Low signal level
	Amplitude	X_{Ampl}	Amplitude of the signal: the difference of high and low signal levels $X_{Ampl} = X_{High} - X_{Low}$
	Max	X_{Max}	Absolute maximum value of the waveform
	Min	X_{Min}	Absolute minimum value of the waveform
	Peak to peak	X_{PkPk}	Peak-to-peak value of the waveform: the difference of maximum and minimum values $X_{Ampl} = X_{Max} - X_{Min}$
	Mean	X_{Mean}	Arithmetic average of the waveform voltage values $X_{Mean} = \frac{1}{N_{Eval}} \sum_{i=1}^{N_{Eval}} x(i)$
	RMS	X_{RMS}	RMS (root mean square, quadratic mean) of the waveform voltage values $X_{RMS} = \sqrt{\frac{1}{N_{Eval}} \sum_{i=1}^{N_{Eval}} x^2(i)}$
	σ (S-dev)	σ_X	Standard deviation of the waveform samples $\sigma_X = \sqrt{\frac{1}{N_{Eval}-1} \sum_{i=1}^{N_{Eval}} (x(i) - X_{Mean})^2}$

	Meas. type	Symbol	Description/Result
	Pos. overshoot	R_{Pos}	Positive overshoot of a square wave, calculated from measurement values High, Max, and Amplitude $+Ovr = \frac{V_{top} - V_{P+}}{V_{Amp}} \cdot 100\%$
	Neg. overshoot	R_{Neg}	Negative overshoot of a square wave, calculated from measurement values Min, Low, and Amplitude $-Ovr = \frac{V_{base} - V_{P-}}{V_{Amp}} \cdot 100\%$
	Cycle mean		The mean value of one cycle
	Cycle RMS		The RMS (root mean square) value of one cycle
	Cycle σ (S-dev)		The standard deviation of one cycle
	Trig. ProbeMeter		The DC voltage from the connected probe. Only available if an active Rohde & Schwarz probe with ProbeMeter is connected.
	Slew rate rising		Steepness of the rising edge: voltage difference between the lower and higher reference level, divided by the rise time. Result in V/s = V*Hz (blue vertical lines in the picture below).
	Slew rate falling		Steepness of the falling edge: voltage difference between the higher and lower reference level, divided by the fall time. Result in V/s = V*Hz (green vertical lines in the picture below). 

Time measurements

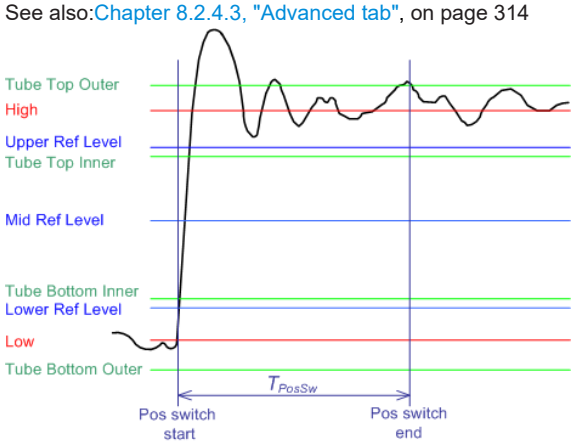
Access: [Measure] > "Setup" tab > "Amp/Time" category

Reference levels are explained in Chapter 8.2.4, "Reference levels", on page 307.

Table 8-2: Time measurements

	Meas. type	Symbol	Description/Result
	Rise time	T_{Rise}	Rise time of the left-most rising edge of the waveform. Rise time is the time it takes the signal to rise from the low reference level to the high reference level. Measurement all events in the acquisition is possible.
	Fall time	T_{Fall}	Falling time of the left-most falling edge of the waveform. Fall time is the time it takes the signal to fall from the high reference to the low reference. Measurement all events in the acquisition is possible.

	Meas. type	Symbol	Description/Result
	Pos. pulse	$T_{PosPulse}$	Width of a positive pulse: time between a rising edge and the following falling edge measured on the middle reference level. The measurement requires at least one complete period of a triggered signal. Measurement all events in the acquisition is possible.
	Neg. pulse	$T_{NegPulse}$	Width of a negative pulse: time between a falling edge and the following rising edge measured on the middle reference level. The measurement requires at least one complete period of a triggered signal. Measurement all events in the acquisition is possible.
	Period	T_{Period}	Time between two consecutive waveform edges of the same direction, measured on the middle reference level. The measurement requires at least one complete period of a triggered signal. Measurement all events in the acquisition is possible.
	Frequency	f_{Period}	Frequency of the signal, reciprocal value of the period. $f_{Period} = 1 / T_{Period}$
	Pos. duty cycle	R_{PosCyc}	Positive duty cycle: Width of a positive pulse in relation to the period in %. The measurement requires at least one complete period of a triggered signal. Multiple measurements are possible. $R_{PosCyc} = \frac{T_{PosPulse}}{T_{Period}} \cdot 100\%$
	Neg. duty cycle	R_{NegCyc}	Negative duty cycle: Width of a negative pulse in relation to the period in %. The measurement requires at least one complete period of a triggered signal. Multiple measurements are possible. $R_{NegCyc} = \frac{T_{NegPulse}}{T_{Period}} \cdot 100\%$
	Delay		The measurement result is negative if the edge of the second source comes before the edge of the first source. See: "Settings for delay and phase measurements (analog sources)" on page 323
	Phase		Phase difference between two waveforms. *Phase = Delay / Period * 360*
	Burst width		Duration of one burst, measured from the first edge to the last

	Meas. type	Symbol	Description/Result
	Pos. switching	T_{PosSw}	Settling time at rising edges: Time between crossing the lower reference level and the last return of the signal into the top tolerance tube. See also:Chapter 8.2.4.3, "Advanced tab", on page 314 
	Neg. switching	T_{NegSw}	Settling time at falling edges: Time between crossing the upper reference level and the last return of the signal into the bottom tolerance tube. See also "Pos. switching" above.
	Pulse train		Duration of N positive pulses, measured from the rising edge of the first pulse to the falling edge of the N-th pulse. Define N for the measurement.
	Setup Hold Setup/Hold time	T_{Setup} and T_{Hold}	Setup and Hold time measurement with positive and/or negative clock edge. See: "Setup/Hold measurement settings" on page 324
	Setup/Hold ratio	$T_{Setup} / (T_{Setup} + T_{Hold})$	Setup/Hold ratio measurement with positive and/or negative clock edge. See: "Setup/Hold measurement settings" on page 324
	Delay to trigger		Time between the trigger event and a following signal slope. High accuracy even if the trigger event is outside the acquisition data. See: • "Delay to trigger measurement settings" on page 326

Area measurements

Access: [Measure] > "Setup" tab > "Amp/Time" category

Area measurements are voltage over time measurements.

Table 8-3: Area measurements

	Meas. type	Symbol	Description/Result
	Area	A_{Ref}	Area between the waveform and a reference level ("Area level", X_{Ref}). $A_{Ref} = \frac{T_{Eval}}{N_{Eval}} \cdot \sum_{i=1}^{N_{Eval}} (x(i) - X_{Ref})$ T_{Eval}: Evaluation time, time of a full waveform or limited by a gate
	Cycle area	A_{RefCyc}	Area between the waveform and a reference level ("Area level") measured for one period, see also "Area" measurement. The measurement requires at least one complete period of a triggered signal. Multiple measurements are possible.

Counting

Access: [Measure] > "Setup" tab > "Amp/Time" category

Table 8-4: Counting measurements

	Meas. type	Symbol	Description/Result
	Pulse count		The number of positive or negative pulses of the waveform, or of both positive and negative pulses. The mean value of the signal is determined. If the signal passes the mean value, an edge is counted. A positive pulse is counted if a rising edge and a following falling edge are detected. A negative pulse is counted if a falling edge and a following rising edge are detected.
	Edge count		The number of positive or negative edges, or of both positive and negative edges. The instrument determines the mean value of the signal and counts an edge every time the signal passes the mean value.

Arithmetic

Access: [Measure] > "Setup" tab > "Amp/Time" category

Table 8-5: Arithmetic measurements

	Meas. type	Symbol	Description/Result
	Arithmetic		Allows you to perform basic arithmetic operations between selected measurement values and/or constant.

8.2.5.2 Settings for amplitude/time measurements

Access: [Measure] > "Setup" tab > "Amp/Time" category

Amplitude and time measurements are available for sources in the time domain. For some amplitude/time measurements, such as delay, setup/hold and delay to trigger, further setting are required to get a measurement result.

If further settings are available for a measurement, a settings icon is shown beside the measurement's name in the "Active measurements" list. Tap the icon to configure the measurement.

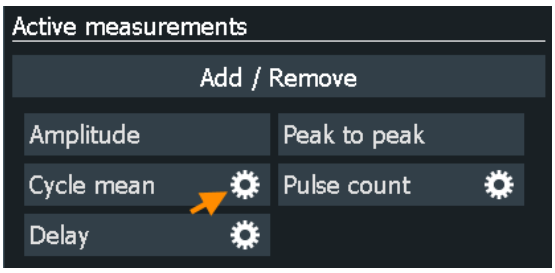


Figure 8-7: Selected active measurements, some have further settings

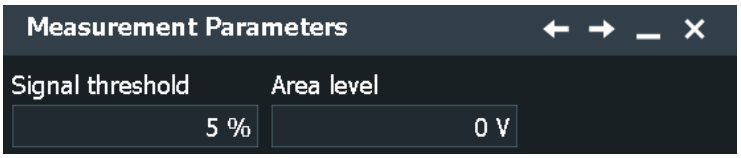


Figure 8-8: Settings for cycle area measurement

This chapter explains all settings for amplitude/time measurements. For a description of available measurement, see Chapter 8.2.5.1, "Overview of amplitude/time measurements", on page 316.

Signal threshold 322
Area level 322
Pulses slope 323
Measured slope 323
Settings for delay and phase measurements (analog sources) 323
Settings for delay and phase measurements (digital sources) 324
Setup/Hold measurement settings 324
Clock ref level 325
Ref level 326
Pulse count 326
Edges slope 326
Delay to trigger measurement settings 326

Signal threshold

Defines a signal value that must be exceeded for the signal value to be included in the measurement. The setting is relevant for area, time, and counting measurements.

Remote command:
`MEASurement<m>:DETThreshold` on page 1493

Area level

The reference level used to integrate the waveform. The setting is only relevant for area measurements.

Remote command:
`MEASurement<m>:AMPTime:ALEVel` on page 1493

Pulses slope
Sets the first slope of the pulses to be counted.

The setting is available only for the "Pulse count" measurement.

"Positive"	Positive pulses are counted.
"Negative"	Negative pulses are counted.
"Either"	Both positive and negative pulses are counted.

Remote command:
`MEASurement<m>:AMPTime:PSLope` on page 1494

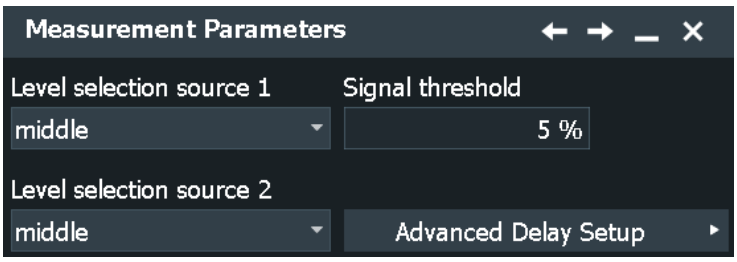
Measured slope
Selects the slope direction for frequency and period measurements.

"Positive / Negative"	Measures the time between rising or falling edges, respectively.
"Either"	In multiple measurements, the time is measured both between rising edges and between falling edges. In single measurements. The first edge is taken for the measurement.
"First edge"	Time is measured either between rising edges or between falling edges. The first edge is taken for the measurement. In single measurements, it works the same way as "Either". Only available for analog measurement sources.

Remote command:
`MEASurement<m>:AMPTime:PFSLope` on page 1493

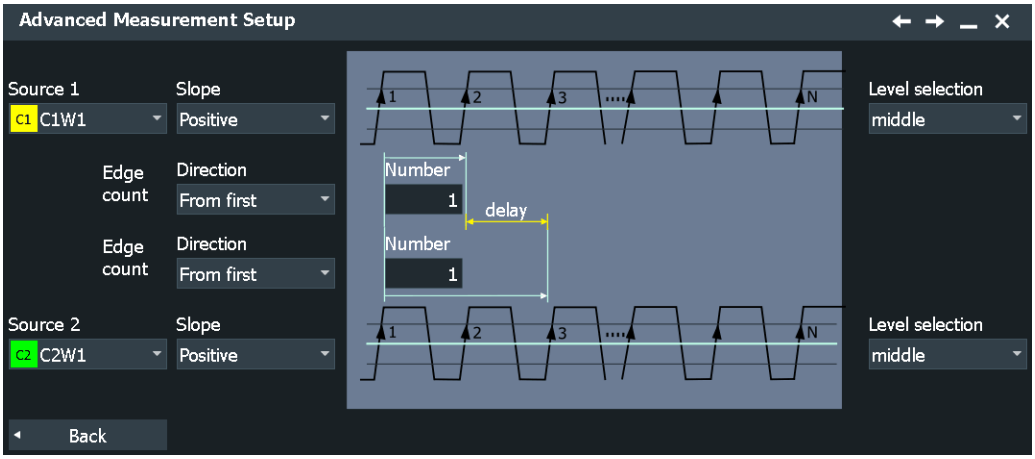
Settings for delay and phase measurements (analog sources)
The specific settings for delay measurement allow you to measure the time between any two slopes at any reference level. Therefore, the reference levels and the slopes must be defined for each source individually. The measurement result is negative if the edge of the second source comes before the edge of the first source.

"Level selection source 1/2": For each measurement source, select the reference level on which the time is measured.



"Advanced Delay Setup":

With the settings shown in the picture, the time between the second and the fifth rising edge is measured.



"Level selection"	Selects the reference level on which the time is measured.
"Slope"	Sets the edge of each source, between which the delay is measured: positive, negative, or either of them.
"Direction"	Selects the direction for counting slopes for each source: from the beginning of the waveform, or from the end.
"Number"	Sets the number of the edge that is relevant for delay measurement.

Remote command:
`MEASurement<m>:AMPTime:DELay<n>:LSELect` on page 1495
`MEASurement<m>:AMPTime:DELay<n>:SLOPe` on page 1495
`MEASurement<m>:AMPTime:DELay<n>:DIRection` on page 1494
`MEASurement<m>:AMPTime:DELay<n>:ECOunt` on page 1495

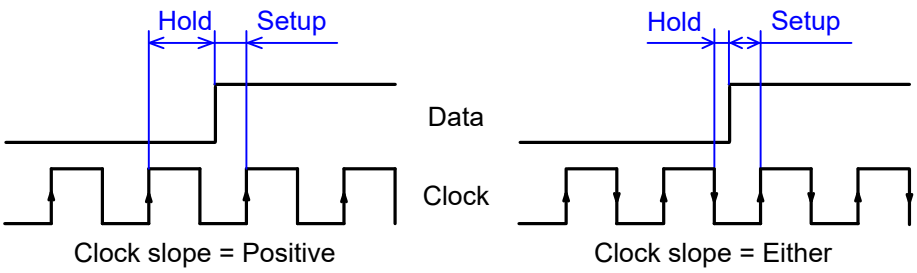
Settings for delay and phase measurements (digital sources)
Delay measurement on digital channels is reduced to measure the time between two subsequent rising or two subsequent falling edges.

The edge direction is set with Edges slope.

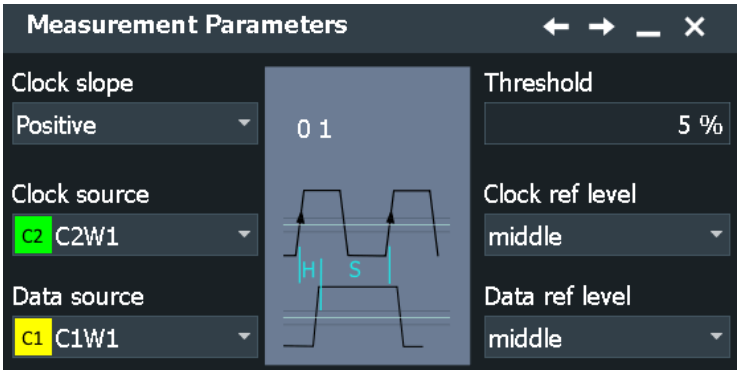
Setup/Hold measurement settings
Setup/Hold measurements analyze the relative timing between two signals: a data signal and the synchronous clock signal. Setup time is the time that the data signal is steady before clock edge - the time between a data transition and the next specified clock edge. Hold time is the time that the data signal is steady after clock edge - the time between a data transition and the previous specified clock edge.

"Setup/Hold time" measures and displays the setup and hold durations. "Setup/Hold ratio" measurements return the ratio of the setup time to the sum of hold and setup time: $T_{Setup} / (T_{Setup} + T_{Hold})$.

The clock edge can be defined, the polarity of the data signal does not matter.



If at least one of the setup/hold measurements is selected, more settings appear to specify the measurement.



"Clock slope" Sets the edge of the clock from which the setup and hold times are measured: positive, negative, or either of them. If "Either" is selected, the clock edges next to the data edge are considered regardless of the clock slope.

"Clock source" The "Clock source" is identical to the measurement "Source". It defines the waveform used as clock in the setup/hold measurement.

"Data source" The "Data source" is identical to the "2nd Source" of the measurement. It sets the data signal.

"Clock ref level" See "Clock ref level" on page 325.

"Data ref level" See "Ref level" on page 326.

"Threshold" See "Signal threshold" on page 322.

Remote command:
Clock slope: `MEASurement<m>:AMPTime:CSLope` on page 1496

Clock ref level

Selects the reference level of the clock on which the time is measured. The intersection of slope and reference level defines the time point for measurements.

The setting is used for setup and hold measurements, and for jitter measurements (option R&S RTO6-K12).

Remote command:
`MEASurement<m>:AMPTime:CLCK<n>:LSELect` on page 1497

Ref level
The setting defines the "Data ref level" for setup and hold measurements, and for time-interval error measurements (option R&S RTO6-K12). It selects the reference level of the data signal on which the time is measured. The intersection of slope and reference level defines the time point for measurements.

For clock jitter measurements (option R&S RTO6-K12), it sets the "Reference level" for the time measurement.

Remote command:
`MEASurement<m>:AMPTime:DATA<n>:LSELect` on page 1497

Pulse count
Sets the number N of positive pulses for the "Pulse train" measurement. This measurement measures the duration of N positive pulses from the rising edge of the first pulse to the falling edge of the N-th pulse.

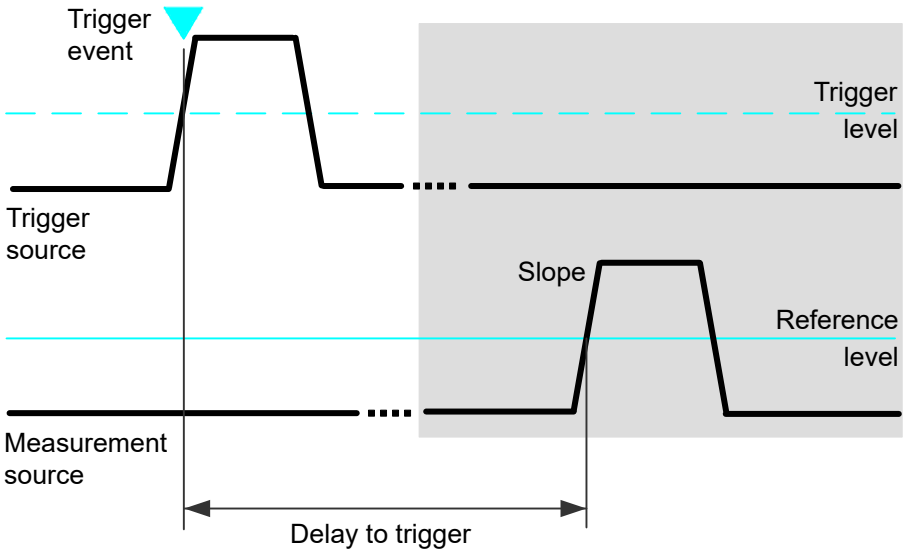
Remote command:
`MEASurement<m>:AMPTime:PTCount` on page 1496

Edges slope
Sets the edge direction to be considered. The setting is relevant for edge count measurement and delay measurement on digital channels.

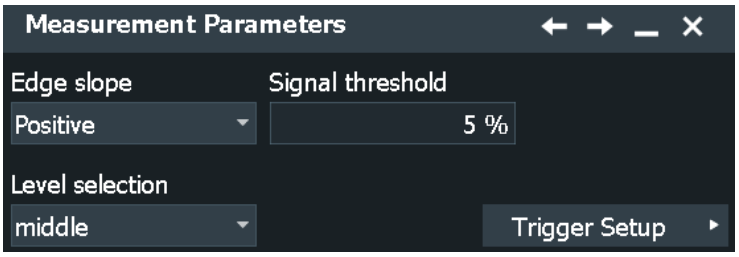
"Positive"	Positive edges are considered.
"Negative"	Negative edges are considered.
"Either"	Both positive and negative edges are counted (edge count). Delay is measured either between rising edges or between falling edges. The first edge is taken for the measurement.

Remote command:
`MEASurement<m>:AMPTime:ESLope` on page 1496

Delay to trigger measurement settings
Delay to trigger measures the time between the trigger point and the following slope of a waveform. The delay between the trigger and the slope can be high compared to the accuracy of the acquisition, and the trigger point can even be outside of the current acquisition.



To configure the trigger conditions, use the trigger setup. To set up the slope, additional settings appear in the measurements "Setup" dialog box.



"Edge slope" Sets the edge direction to be used for delay measurement: positive, negative, or either edge.

"Level selection" Selects the reference level of the measurement source on which the delay is measured: upper, middle, or lower level.

"Signal threshold"
See "Signal threshold" on page 322.

Remote command:
`MEASurement<m>:AMPTime:DTOTrigger<n>:SLOPe` on page 1497
`MEASurement<m>:AMPTime:DTOTrigger<n>:LSELect` on page 1498

8.2.5.3 Arithmetic for amplitude/time and spectrum measurements

Access: [Measure] > "Setup" tab > "Amp/Time"/"Spectrum" category > "Arithmetic".

In this dialog, you can set up basic arithmetic operations between selected measurement values and/or constant.

You can select only measurement groups and measurements that are already enabled.

Type
Selects the type of arithmetic measurement:

"Measurement" Performs an arithmetic operation between two measurement results. You can select from the measurements enabled for "Amp/Time" or "Spectrum".

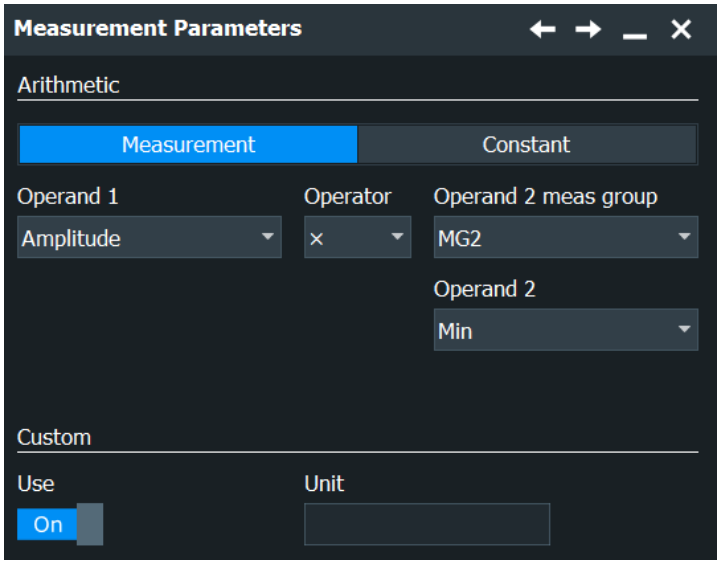


Figure 8-9: Arithmetic with two measurement results

"Constant" Performs an operation between a measurement result and a user defined fixed Constant.

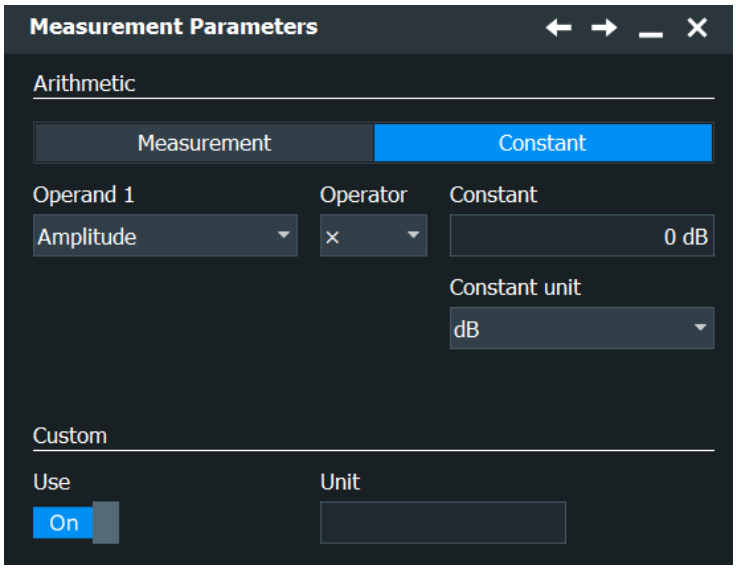


Figure 8-10: Arithmetic with a measurement result and a constant

Remote command:
`MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:TYPE` on page 1504

Operand 1
Selects the first operand for the arithmetics. You can select one of the measurements, that is enabled for the current measurement group.

Remote command:
`MEASurement<m>:SPECtrum:ARIThmetic:PARameter<n>:AMPTime` on page 1506
`MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:AMPTime` on page 1502
`MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:SPECtrum` on page 1503
`MEASurement<m>:SPECtrum:ARIThmetic:PARameter<n>:SPECtrum` on page 1507

Operand 2 meas group / Operand 2
Selects the measurement group and the measurement result for the second operand. You can select one of the measurements, that is enabled for the selected measurement group.

Remote command:
`MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:GROup` on page 1503
`MEASurement<m>:SPECtrum:ARIThmetic:PARameter<n>:GROup` on page 1507

Operator
Selects the operation type to be performed on the selected operands. You can choose between +, -, * and /.

"+"	Adds up the operands.
"-"	Subtracts "Operand 2"/"Constant" from "Operand 1".
"*"	Multiplies the operands.
"/"	Divides "Operand 1" by "Operand 2"/"Constant".

Remote command:
`MEASurement<m>:AMPTime:ARIThmetic:OPERator` on page 1502

Custom
Enable "Use" to apply the user-defined value for the "Unit".

Remote command:
`MEASurement<m>:AMPTime:ARIThmetic:CUNState` on page 1501
`MEASurement<m>:SPECtrum:ARIThmetic:CUNState` on page 1505

Constant
Sets a constant numeric value for the arithmetic operation, for the "Constant" measurement type.

Remote command:
`MEASurement<m>:AMPTime:ARIThmetic:CONStant` on page 1500
`MEASurement<m>:SPECtrum:ARIThmetic:CONStant` on page 1504

Constant unit
Selects the unit for the Arithmetic for amplitude/time and spectrum measurements.

Remote command:
`MEASurement<m>:AMPTime:ARIThmetic:COUNit` on page 1500
`MEASurement<m>:SPECtrum:ARIThmetic:COUNit` on page 1505

8.2.5.4 Measuring the delay to trigger

Delay to trigger measures the time between the trigger point and the following slope of a waveform. If the delay is unknown, it can be measured in two stages - first a coarse and then a precise measurement.

See also: "Delay to trigger measurement settings" on page 326.

1. Set the horizontal scale and horizontal position so that the trigger point and the slope both are visible on the screen.
2. Select the delay to trigger measurement:
   a) In the "Setup" dialog, set the "Category" to "Amp/Time".
   b) Press "Add/ Remove Measurements" button.
   c) In the dialog, enable "Delay to trigger".
   d) Tap "Back".
3. Configure the "Delay to trigger" measurement:
   a) In the "Active Measurements" list, tap the "Delay to trigger" measurement.
   b) Select the source, that is the waveform with the delayed slope.
   c) Select the slope and the reference level.
   d) Check the trigger settings.
4. Enable the measurement. Note the result.
5. Turn the horizontal [Position] knob and enter the measured delay as horizontal position.

   Thus, the slope is moved to the center of the screen.
6. Adjust the time scale ("Menu" > "Horizontal" > "Setup" tab) and the sample rate and resolution ("Menu" > "Acquire" > "Setup" tab) to the required resolution.

   The trigger is outside the display and is not part of the current acquisition.
7. Repeat the "Delay to trigger" measurement.

   Now the delay is measured with high accuracy. You can analyze the variance of delay values using statistical evaluation and histogram functions.

8.2.6 Eye diagram measurements

The eye diagram is a tool for evaluation of signal quality and shows the combined effects of channel noise and intersymbol interference. It is a significant means of visualizing jitter and allows you to analyze the reasons for it. By creating histograms of the eye diagram, important jitter parameters can be determined.

The eye diagrams are a superposition of repetitively sampled waveforms, which have a length of about 1 bit.

There are two ways to generate eye diagrams on the R&S RTO6:

- Standard eye diagrams, which are a superposition of repetitively sampled digital data.
- Advanced eye diagrams, which are generated by superimposing waveform slices from a multi-bit acquisition. To use advanced eye diagrams, option R&S RTO6-K91 is required.

The waveform display style must be set to vectors: "Menu" > "Settings" > "Display" > "Signal" > "Waveform style" = "Vectors".

To obtain optimized settings for an eye measurement, use the "Autoset" function that is provided on the right side of the "Eye" tab.

The following characteristic values can be determined:

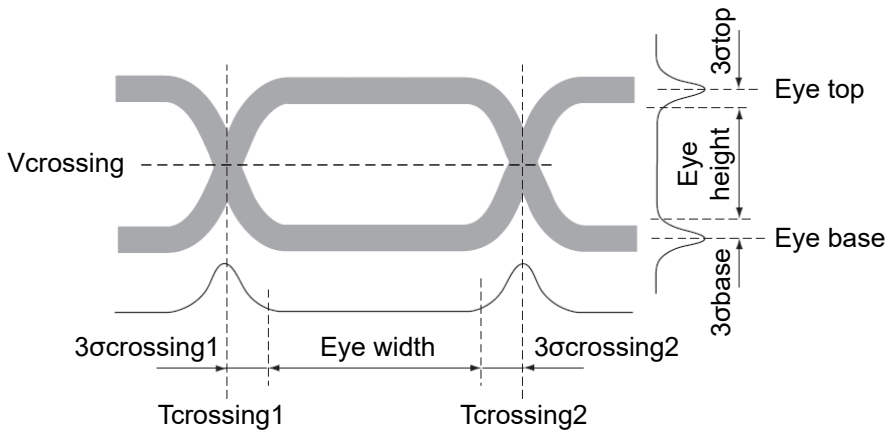


Figure 8-11: Basic eye diagram characteristics

Eye top = Mean of the upper vertical histogram
σtop = Standard deviation of the upper vertical histogram
Eye base = Mean of the lower vertical histogram
σbase = Standard deviation of the lower vertical histogram
Tcrossing = First and second mean of the horizontal histogram
σcrossing = Standard deviation of the horizontal histogram

8.2.6.1 Overview of eye diagram measurements

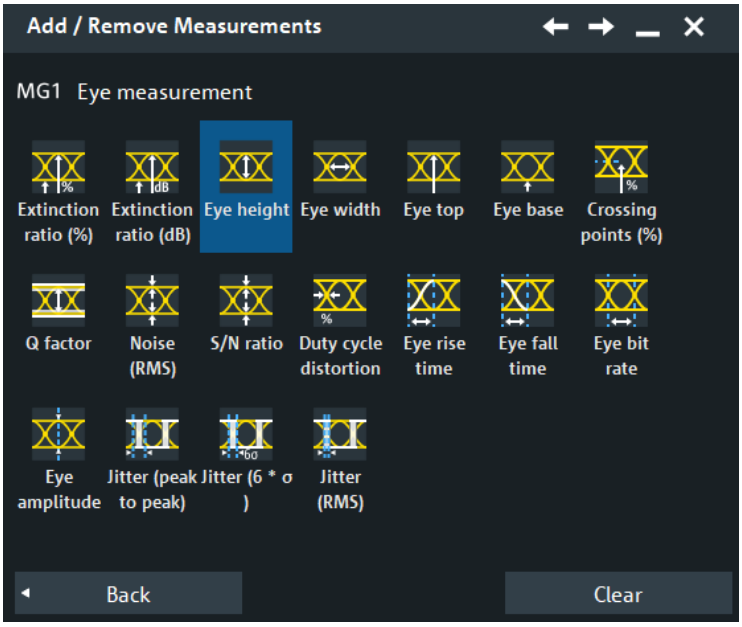


Table 8-6: Eye measurements

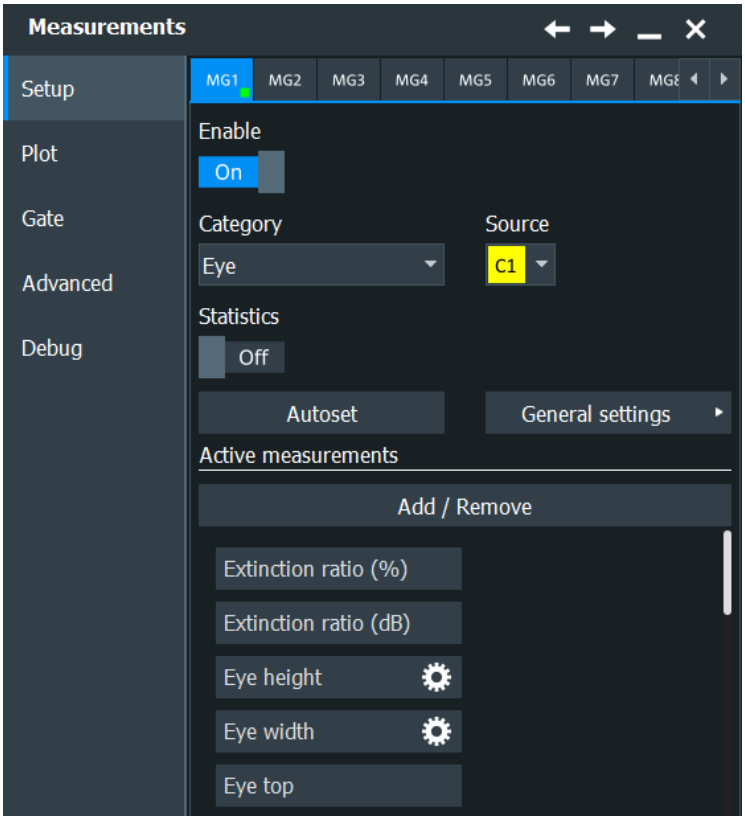
	Meas. type	Description/Result
	Extinction ratio (%)	The extinction ratio is an indication of efficiency. It describes the ratio of the power used to transmit a logic level 1, to the energy used to transmit a logic level 0. The R&S RTO6 provides extinction ratio measurements as a percentage, and in decibels: ER (%) = Eye base / Eye top *100 Prerequisite: Eye base > 0 and Eye top > 0 because extinction ratio is defined only for positive values.
	Extinction ratio (dB)	ER (dB) = 10*log (Eye top / Eye base)
	Eye height	The vertical eye opening indicates the sensitivity of the transmission to noise. (Eye top – 3 * σtop) – (Eye base + 3 * σbase)
	Eye width	The horizontal eye opening indicates the time range during which the sampling of the logical state is possible. (Tcrossing2 – 3 * σcrossing2) – (Tcrossing 1 – 3 * σcrossing1)
	Eye top	Mean of the upper vertical histogram
	Eye base	Mean of the lower vertical histogram
	Q factor	Q factor = (Eye top – Eye base) / (σtop + σbase)

	Meas. type	Description/Result
	Noise (RMS)	Quadratic mean of the noise of eye top and eye base Noise (RMS) = (σtop + σbase)/2
	S/N ratio	Signal-to-noise ratio SNR = 10 * log (Eye amplitude / Noise RMS)
	Duty cycle distortion	Duty cycle distortion = 20 * log (Eye amplitude / Noise RMS)
	Eye rise time	Duration for signal to rise from 10% to 90% of the high signal level
	Eye fall time	Duration for signal to fall from 90% to 10% of the high signal level
	Eye bit rate	Frequency between two crossings
	Eye amplitude	Eye top - Eye base
	Jitter (peak to peak)	Average of the jitter for both crossing points. Jitter = (σcrossing1 + σcrossing2) / 2
	Jitter (6*σ)	Jitter (6*σ) = Jitter * 6
	Jitter (RMS)	Quadratic mean of the jitter at both crossing points
	Crossing points (%)	Percentage of points crossing the eye

8.2.6.2 Settings for eye diagram measurements

Access: [Measure] > "Setup" tab > "Eye" category

Eye diagram measurements are only available for sources in the time domain.



To obtain optimized settings for an eye measurement, use the "Autoset" function that is provided on the right side of the tab.

Autoset
Defines optimized settings for eye diagram measurements on the selected source.

Remote command:
`MEASurement<m>:EYEJitter:AUToscale` on page 1509

8.2.7 Spectrum measurements

Spectrum analysis determines the frequencies of a given input signal over time. Various measurements can then be performed based on the signal spectrum.

8.2.7.1 Overview of spectrum measurements

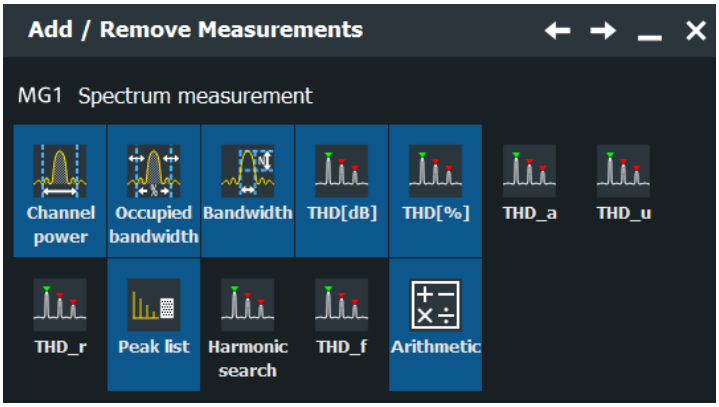


Table 8-7: Spectrum measurements

	Meas. type	Description, result
	Channel power	Power integrated over the sample values defined by a center frequency and a bandwidth; based on a defined impedance. The result is given in dBm. To get best results, enable "Measure all events in each acquisition" on page 362, and set the "Limit" to the maximum value. In particular, these settings are important when measuring pulsed signals.
	Occupied bandwidth	From the defined center frequency, symmetric sample value pairs to the left and right are integrated until a user-defined percentage of the total power is reached. The occupied bandwidth is the difference between the frequencies at which the requested power was reached.
	Bandwidth	n dB down bandwidth; the samples to the left and right of the peak value are analyzed until the n dB threshold is exceeded. The frequencies at which the threshold is exceeded define the limits of the requested bandwidth.
	THD[dB], THD[%] Total harmonic distortion	Power sum of the harmonic waves divided by the power of the fundamental wave: $THD = \frac{\sum_{n=2}^{\infty} P_n}{P_1}$

	Meas. type	Description, result
	THD_f, THD_a, THD_u, THD_r Total harmonic distortion	These measurements require option R&S RTO6-K37 Spectrum analysis. THD_f is the root mean square of the sum of all amplitudes of the harmonic waves in relation to the amplitude at the fundamental frequency (first harmonic): $THD_F = \frac{\sqrt{\sum_{i=2}^{n} U_i^2}}{U_1}$ THD_a corresponds to THD[dB]: $THD_a = \frac{\sum_{i=2}^{n} U_i^2}{U_1^2}$ THD_u: $THD_u = \frac{\sqrt{U^2 - U_1^2}}{U_1}$ Distortion factor: $THD_R = \frac{\sqrt{U^2 - U_1^2}}{U}$ Where: • U_i: effective value of the harmonic with index i • U_1: effective value of the first harmonic • U: effective value of the signal
	Peak list	This measurement requires option R&S RTO6-K37 Spectrum analysis. Table with measured peaks. For each peak, the frequency and the power value are listed in a table row. The number of determined peaks can be defined. You can sort the results by frequency or power value, and the peak labels are adjusted accordingly.
	Harmonic search	Table with measured harmonics. For each harmonic, the frequency and the value are listed in a table row.
	Arithmetic	Allows you to perform basic aritmetic operations between selected measurement values and/or constant. For details see: Chapter 8.2.5.3, "Arithmetic for amplitude/time and spectrum measurements", on page 327

For remote command parameters and suffix types, see Table 24-8.

8.2.7.2 Settings for spectrum measurements

Access: [Measure] > "Setup" tab > "Spectrum" category

Spectrum measurements require a source in the frequency domain, i.e. a math waveform with an FFT operation.

For spectrum measurements, make sure that the start frequency and other FFT parameters are set correctly, and the fundamentals are not covered by the DC component of the signal. Consider also a gated measurement if the instrument cannot return any result.

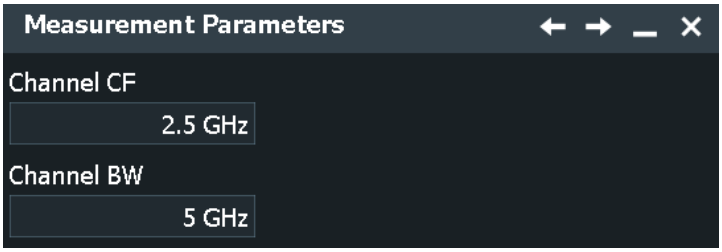


Figure 8-12: Parameters of channel power measurement

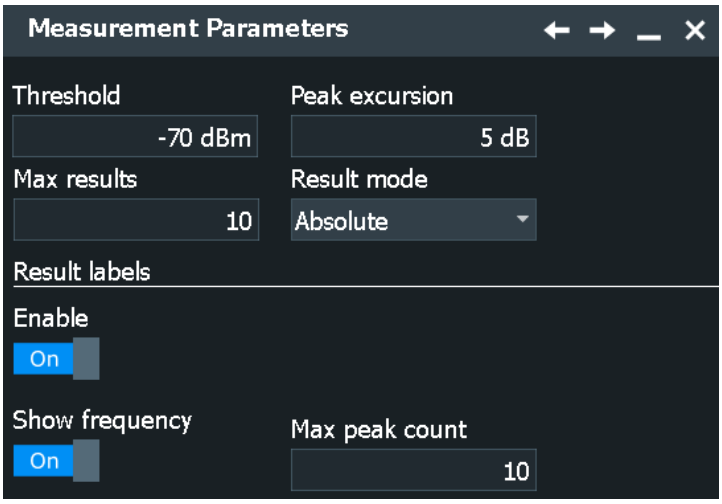


Figure 8-13: Parameters of peak list measurement (with option R&S RTO6-K37)

Enable
Enables the peak list measurement for the spectogram.

N db down
The threshold until which the samples to the left and right of the peak value are analyzed to determine the "Bandwidth".

Remote command:
`MEASurement<m>:SPECtrum:NDBDown` on page 1513

Channel BW
Bandwidth over which the channel power is calculated.

Remote command:
`MEASurement<m>:SPECtrum:CPOWer:BANDwidth` on page 1512

Channel CF
Center frequency from which the channel power is calculated over the specified bandwidth.

Remote command:
`MEASurement<m>:SPECtrum:CPOWer:CFRequency` on page 1513

Occup. BW
Percentage of the total power used to determine the occupied bandwidth.

Remote command:
`MEASurement<m>:SPECtrum:OBANdwidth` on page 1513

Threshold
Defines an absolute threshold as an additional condition for the peak search. Only peaks that exceed the threshold are detected.

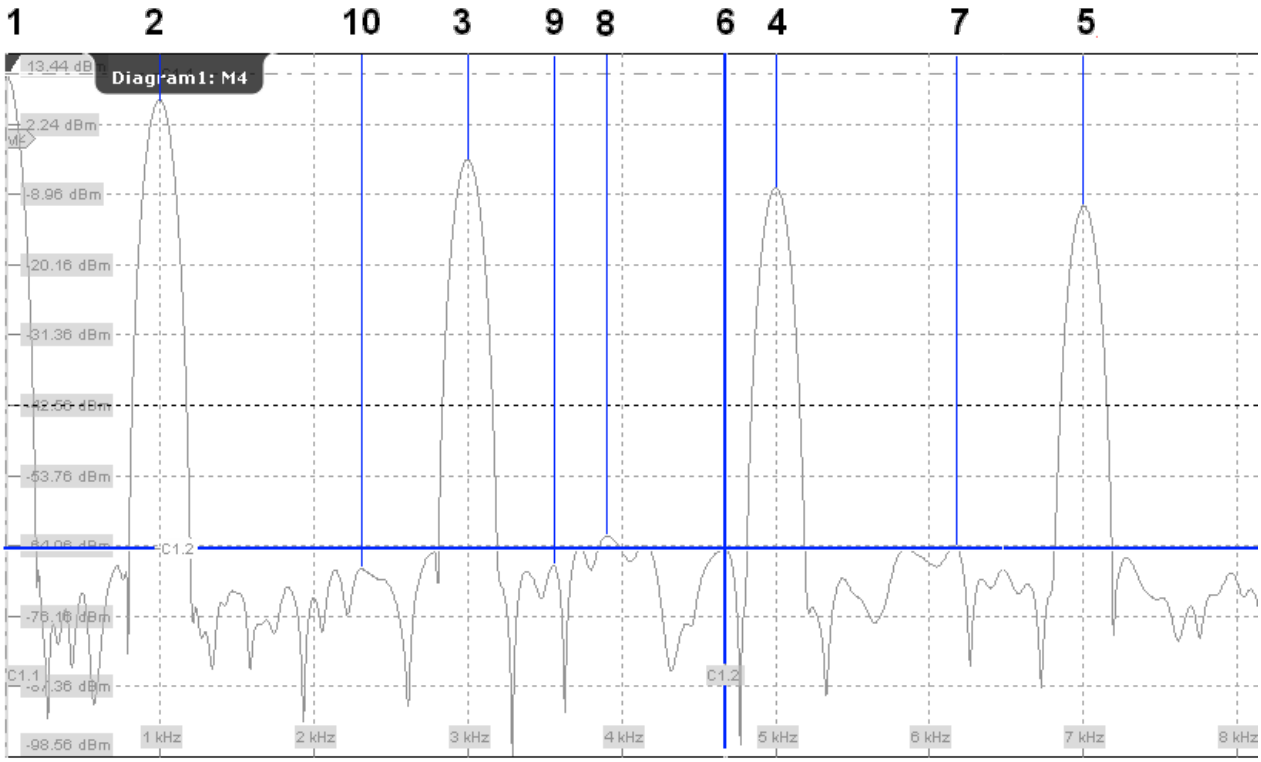
This setting is only available for spectrum waveforms. It is valid for cursor measurements, spectrum measurements and peak search.

Remote command:
`CURSor<m>:THReshold` on page 1631
`MEASurement<m>:SPECtrum:ATHReshold` on page 1514

Peak excursion
Defines a relative threshold, the minimum level value by which the waveform must rise or fall to be considered as a peak. To avoid identifying noise peaks, enter a peak excursion value that is higher than the noise levels.

This setting is only available for spectrum waveforms. It is valid for cursor measurements, spectrum measurements and peak search.

The following figure shows a cursor measurement on a spectrum waveform:



If "Peak excursion" is 30 dB, the peaks 1 to 5 are found. If "Peak excursion" is 20 dB, also the peaks 6 to 10 are found. The cursor position is on peak 6.

Remote command:
`CURSor<m>:PEXCursion` on page 1630
`MEASurement<m>:SPECtrum:PEXCursion` on page 1514

Max results
Sets the maximum number of measurement results that are listed in the result table. Available for peak list and harmonic search measurements.

Remote command:
`MEASurement<m>:SPECtrum:RESult<n>:COUNt` on page 1514

Result mode
Selects the the way the measurement results are displayed. Available for peak list and harmonic search measurements.

"Absolute"	The harmonics/peaks are shown in absolute value, dBm.
"Relative"	The level of the carrier is shown in absolute value dBm. The values the harmonics/peaks are shown relatively to the carrier in dBc.

Remote command:
`MEASurement<m>:SPECtrum:RESult<n>:MODE` on page 1515

Result labels
For peak lists only, requires option R&S RTO6-K37: Use labels to describe the detected peaks in the spectrum diagram.

You can configure what is displayed in the labels. To change the label design, use "Menu" > "Settings" > "Appearance" > "Peak List", see Chapter 4.3.9, "Peak list appearance settings", on page 87.

"Enable"	Displays a description for each detected peak in the spectrum diagram.
"Max. peak count"	Defines the maximum number of peaks that are labeled in the diagram. The result table lists all peaks.
"Show Frequency"	Includes the frequency of the detected peak in the diagram labels.

Remote command:
`MEASurement<m>:RESult:SHLabels` on page 1516
`MEASurement<m>:RESult:LABorder` on page 1315
`MEASurement<m>:RESult:INVerse` on page 1316
`MEASurement<m>:RESult:MAXCount` on page 1515
`MEASurement<m>:RESult:SHFRequency` on page 1516

8.2.8 Histograms and histogram measurements

8.2.8.1 Histogram characteristics

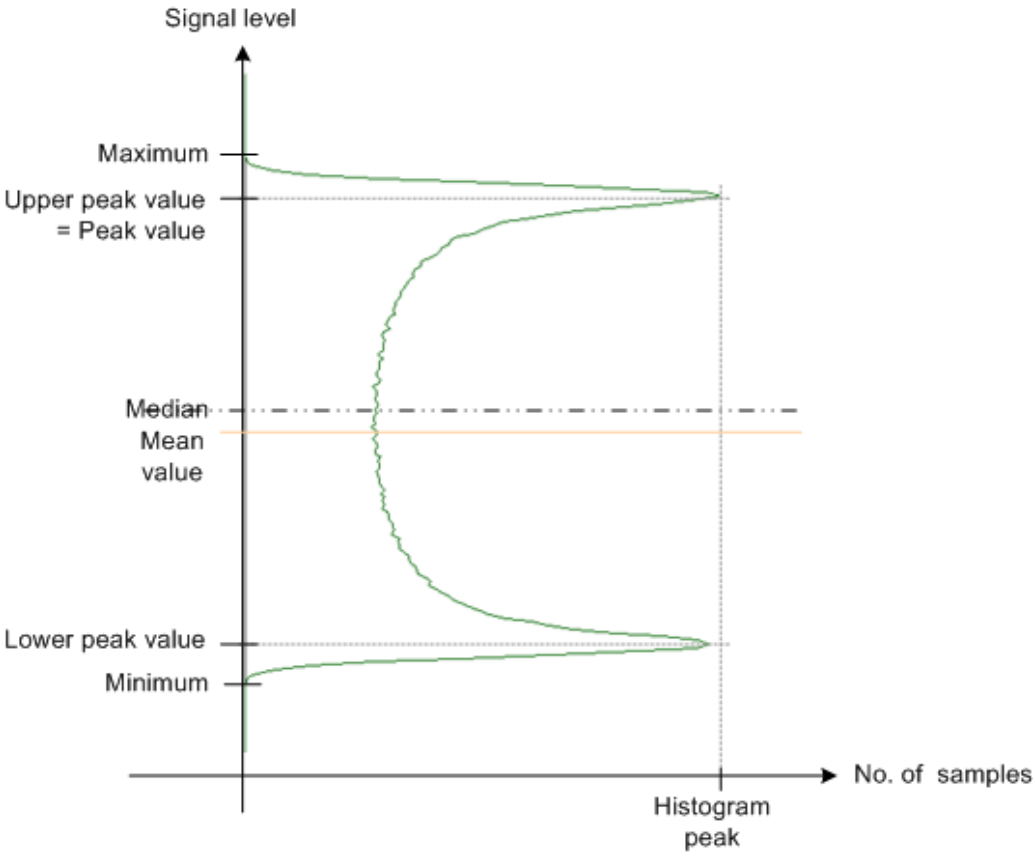
Histograms are used to plot density of data, i.e. to display graphically how often which signal values occur. The histogram can be based on the input signal levels (amplitudes) or the timebase in a time domain measurement, or on frequencies or frequency levels in a spectrum measurement. They are a prerequisite for histogram measurements.

Depending on which data the histogram is based on, a vertical or horizontal histogram can be selected. A vertical, or amplitude, histogram displays horizontal bars across amplitude values. A horizontal or time/frequency histogram displays vertical bars over time/frequencies.

You can define up to 8 histograms in a diagram, one of them is displayed. They can be created quickly using toolbar icons, or in the "Meas" menu > "Histogram" dialog box. To switch the histogram display, tap the required histogram area, or select it in the "Histogram" dialog box. For histogram measurements, the measured histogram is selected independently in the measurement setup.

In a histogram, the maximum count of a waveform value is assigned to the full height (histogram peak). All other count values are displayed relative to the maximum.

The following characteristic values can be determined for histograms (illustrated for a vertical histogram):



8.2.8.2 Overview of histogram measurements

There are various measurements available to analyze histograms.

Note that graphical changes is the diagram, e.g. changing the diagram size or the size of the result table, reset the histogram measurements in run and stop mode.

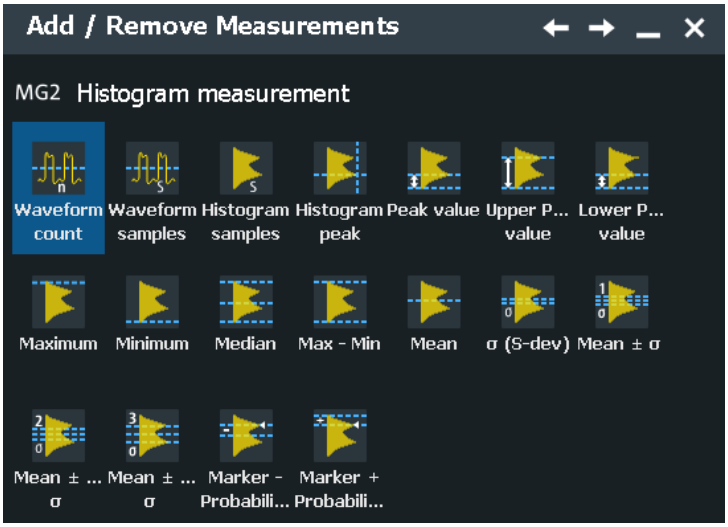


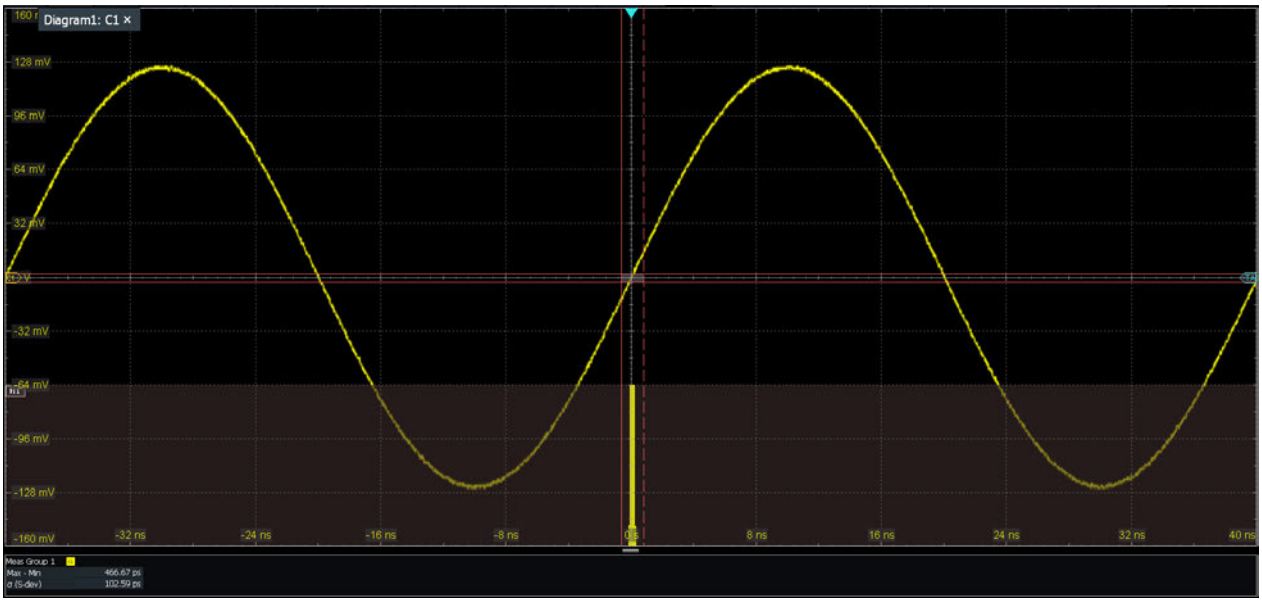
Table 8-8: Histogram measurements

	Meas. type	Description/Result
	Waveform count	The number of acquisitions (waveforms) the histogram is based on
	Waveform samples	The number of samples from the most recent acquisition included in the current histogram
	Histogram samples	The number of samples from all acquisitions included in the current histogram
	Histogram peak	The maximum count value in the histogram
	Peak value	The signal value at the histogram peak
	Upper peak value	The signal value at the maximum count value in the upper half of the histogram
	Lower peak value	The signal value at the maximum count value in the lower half of the histogram
	Maximum	The highest signal value with a probability > 0
	Minimum	The lowest signal value with a probability > 0
	Median	The signal value for which half the samples lie above, the other half below in the histogram The sample count of one signal value after the other are accumulated until half the total number of samples in the histogram is reached. The signal value for which 50% of the samples are accumulated is the median.
	Max - Min	The range of signal values with a probability > 0
	Mean	The weighted arithmetic average of the histogram
	σ (S-dev)	Standard deviation of the sample numbers

	Meas. type	Description/Result
	Mean ±σ	The range between (mean value + standard deviation) and (mean value - standard deviation)
	Mean ±2*σ	The range between (mean value + 3 * standard deviation) and (mean value - 2 * standard deviation)
	Mean ±3*σ	The range between (mean value + 3 * standard deviation) and (mean value - 2 * standard deviation)
	Marker + Probability %	The marker value (according to the selected probability domain marker type) plus the defined limit. Note that the value is restricted to the histogram range.
	Marker - Probability %	The marker value (according to the selected probability domain marker type) minus the defined limit. Note that the value is restricted to the histogram range.

Rough jitter evaluation using a histogram

You can use a horizontal histogram to perform a rough jitter measurement. Define a histogram for a narrow amplitude range close to the trigger time. The "Max-Min" value indicates the peak jitter, while the "StdDev" value indicates the RMS jitter.



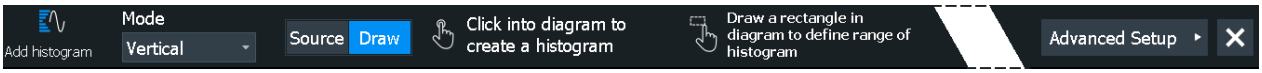
In addition to histograms on channel, math and reference waveforms, histograms can be created based on statistic measurement results. These histograms are enabled in the "Result Analysis" tab, see Chapter 8.2.12, "Result analysis", on page 355.

8.2.8.3 Creating histograms

Histograms can be used to evaluate the sample value occurrences directly. They are a prerequisite for histogram measurements.

To create a histogram quickly with toolbar icons

1. Select the waveform for which you need a histogram.
2. Tap the histogram icon on the toolbar.
3. Select the histogram type on the overlay menu: "Mode" = vertical for an amplitude histogram, or horizontal for a time-based histogram.



4. Tap the diagram with the waveform to be measured, or draw a rectangle on the screen to define the area for histogram calculation.
   Alternatively, tap "Source" and select the waveform.

   The histogram range is indicated in the diagram and a histogram with the selected waveform as a source is defined and displayed.
5. To adjust the histogram range:
   a) Double-tap the histogram.
   b) Tap "Advanced Settings" on the overlay menu.
   c) Adjust the start and stop values.

To create and configure a histogram in the dialog box

1. Select "Menu" > "Apps" > "Analysis" tab > "Histogram".

   The "Histogram" dialog box is displayed.

2. To create a histogram, tap the "Add" icon in the upper right corner of the dialog box.

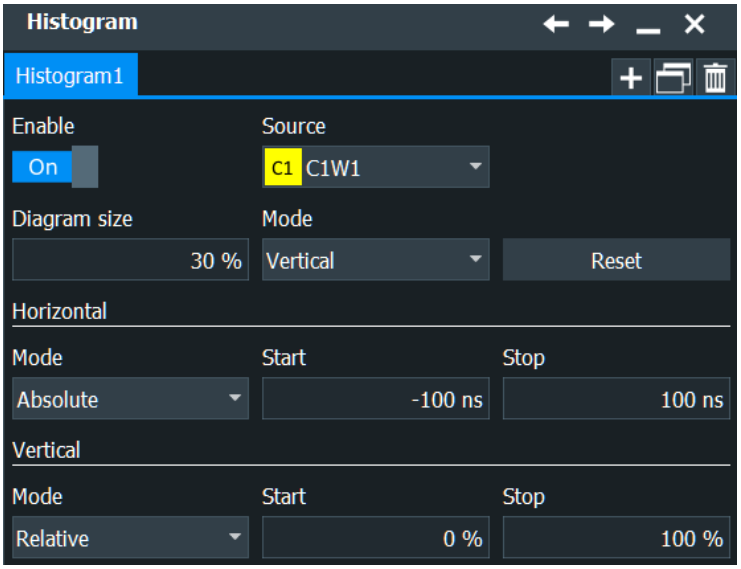
3. To copy an existing histogram and configure a new one based on those settings, tap the "Copy" icon.
4. To change the name of a histogram, double-tap the tab label. Enter a name for the histogram using the on-screen keyboard.
5. Select a "Source" for the histogram. The source can be any input signal, math or reference waveform.
6. Define the histogram "Mode": vertical for an amplitude, horizontal for a time-based histogram.
7. Define the range of the waveform for which the histogram is to be generated. Enter the start value and the stop value in x and in y direction, either as absolute or relative values.

8. Enable the histogram.

8.2.8.4 Histogram setup

"Menu" > "Apps" > "Analysis" tab > "Histogram"

In this dialog box, you configure histograms on which you can perform further measurements.



Enable
Enables or disables the histogram evaluation and display. The histogram settings are kept until the histogram is deleted.

Remote command:
`LAYout:HISTogram:ENABle` on page 1522

Source
Defines the source of the histogram. Any analog channel waveform, math or reference waveform can be selected. Also measurements can serve as histogram source. In this case, the density distribution of the results of the main measurement is displayed.

Remote command:
`LAYout:HISTogram:SOURce` on page 1523

Diagram size
Defines the size of the histogram in percent of the diagram.

Mode
Defines the type of histogram.

"Vertical"	Amplitude histogram (horizontal bars across amplitude)
"Horizontal"	Time histogram (vertical bars over time). For spectrum waveforms, horizontal histograms over spectrum are not available.

Remote command:
`LAYout:HISTogram:MODE` on page 1523

Reset
Resets the values to begin a new histogram.

Remote command:
`LAYout:HISTogram:RESet` on page 1527

Mode (Horizontal / Vertical)
Defines whether the value range limits are entered as absolute or relative values.

Remote command:
`LAYout:HISTogram:HORZ:MODE` on page 1524
`LAYout:HISTogram:VERTical:MODE` on page 1525

Start, Stop (Horizontal)
Defines the horizontal value range of the histogram.

Remote command:
`LAYout:HISTogram:HORZ:ABSolute:STARt` on page 1524
`LAYout:HISTogram:HORZ:ABSolute:STOP` on page 1524
`LAYout:HISTogram:HORZ:RELative:STARt` on page 1525
`LAYout:HISTogram:HORZ:RELative:STOP` on page 1525

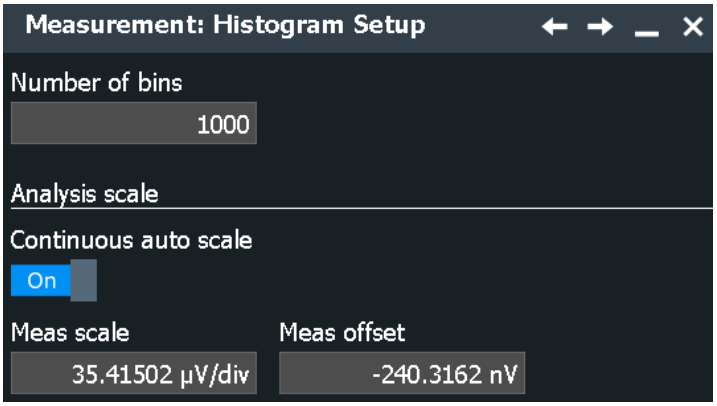
Start, Stop (Vertical)
Defines the vertical value range of the histogram.

Remote command:
`LAYout:HISTogram:VERTical:ABSolute:STARt` on page 1526
`LAYout:HISTogram:VERTical:ABSolute:STOP` on page 1526
`LAYout:HISTogram:VERTical:RELative:STARt` on page 1526
`LAYout:HISTogram:VERTical:RELative:STOP` on page 1526

8.2.8.5 Settings for histogram measurement

Access: [Measure] > "Setup" tab > "Histogram" category

You can perform measurements on histograms. Before, you have to define a histogram, see Chapter 8.2.8.3, "Creating histograms", on page 344.



The measurement parameters are common for all histograms measurements, you find them directly on the "Setup" tab.

Source
Selects the histogram on which the measurement is based. Before, you have to define a histogram, see Chapter 8.2.8.3, "Creating histograms", on page 344.

Remote command:
`MEASurement<m>:HISTogram:SELect` on page 1528

Probability domain marker, Reference
Defines the marker reference in the probability domain.

"Peak"	The y-value with the maximum sample value in the histogram
"Upper Peak"	The y-value at the maximum sample value in the upper half of the histogram
"Lower Peak"	The y-value at the maximum sample value in the lower half of the histogram
"Maximum"	The highest y-value with a probability > 0
"Minimum"	The lowest y-value with a probability > 0
"Median"	The y-value for which half the samples lie above, the other half below in the histogram.
"Mean"	The weighted arithmetic average of the histogram

Remote command:
`MEASurement<m>:HISTogram:PROBability:TYPE` on page 1529

Probability domain marker, Delta
Defines a range around the marker.

Remote command:
`MEASurement<m>:HISTogram:PROBability:LIMit` on page 1529

8.2.9 Jitter measurements

Jitter measurements are availabe if one of the following options is installed:

- R&S RTO6-K12

- R&S RTO6-K133
- R&S RTO6-K134

See Chapter 18.1, "Jitter measurements (options R&S RTO6-K12/K13)", on page 1069.

8.2.10 Optical measurements

This dialog provides settings for testing optical signals. Optical measurements require an optical power signal provided by an optical probe.

For accurate measurements, a dark calibration of the optical probe is required.

8.2.10.1 Overview of optical measurements

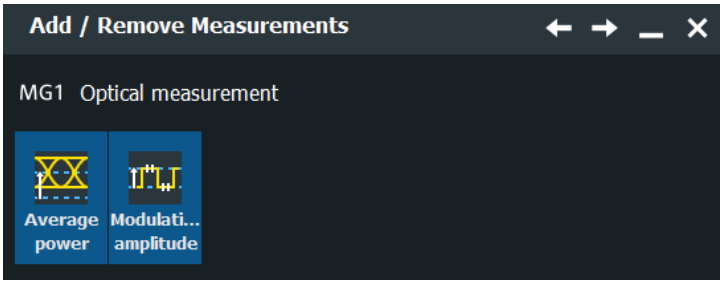


Table 8-9: Optical measurements

	Meas. type	Description/Result
	Average power	Measure for the true average component of an optical signal. $X_{Average} = \frac{1}{N_{Eval}} * \sum_{i=1}^{N_{Eval}} x(i)$
	Optical modulation amplitude (OMA)	Measure of the difference between the optical power of a non-return-to-zero (NRZ) one pulse and the optical power of an NRZ zero pulse. OMA requires an NRZ pattern. The one pulse and zero pulse levels are measured in the level measurement window defined with Start and Stop. OMA is based on norm IEC 61280-2-2.

8.2.10.2 Settings for optical measurements

Access: [Measure] > "Setup" tab > "Optical" category

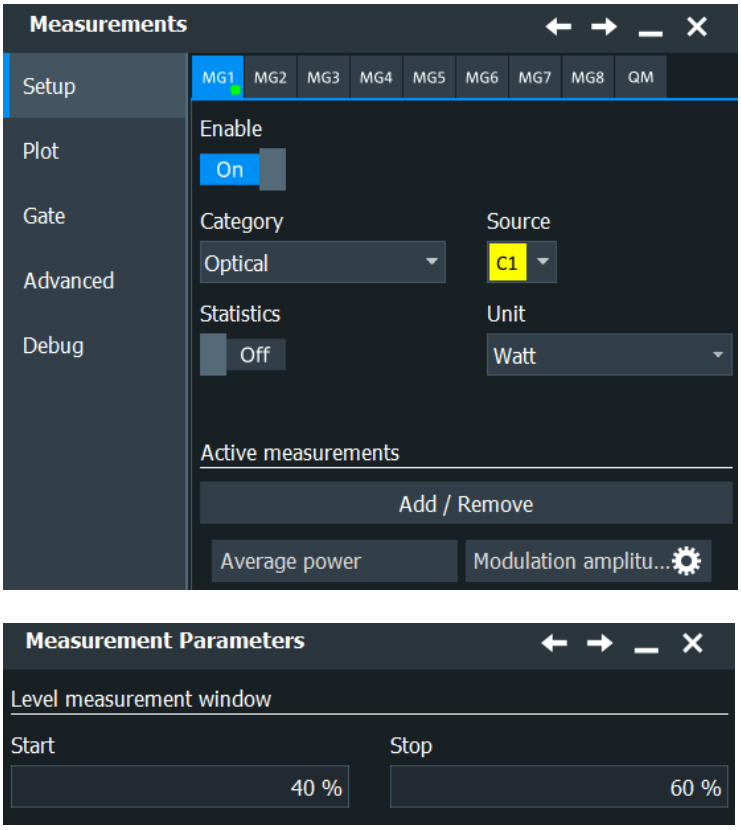


Figure 8-14: Modulation amplitude settings

Unit
Selects the unit of the optical measurement.

Remote command:
`MEASurement<m>:OPTical:UNIT` on page 1519

Start
Sets the start value of the range for the optical modulation amplitude.

Remote command:
`MEASurement<m>:OPTical:OMAMplitude:RELative:STARt` on page 1519

Stop
Sets the end value of the range for the optical modulation amplitude.

Remote command:
`MEASurement<m>:OPTical:OMAMplitude:RELative:STOP` on page 1519

8.2.11 Protocol measurements (option R&S RTO6-K500)

Option R&S RTO6-K500 provides specific measurements on automotive and Ethernet protocols.

8.2.11.1 Overview of protocol measurements

If option R&S RTO6-K500 is installed, enhanced measurements on the supported serial buses are possible. With it, you can measure various dependencies between the frames.

Additional to option R&S RTO6-K500 you need one of the following serial protocol options:

- I2C (R&S RTO6-K510)
- SPI (R&S RTO6-K510)
- UART/RS232 (R&S RTO6-K510)
- CAN / CAN-FD (R&S RTO6-K520/R&S RTO6-K520)
- LIN (R&S RTO6-K520)
- RFFE (R&S RTO6-K550)
- SENT (R&S RTO6-K520)
- Ethernet (100BASE-Tx) (R&S RTO6-K540)
- 100BASE-T1 (R&S RTO6-K560)

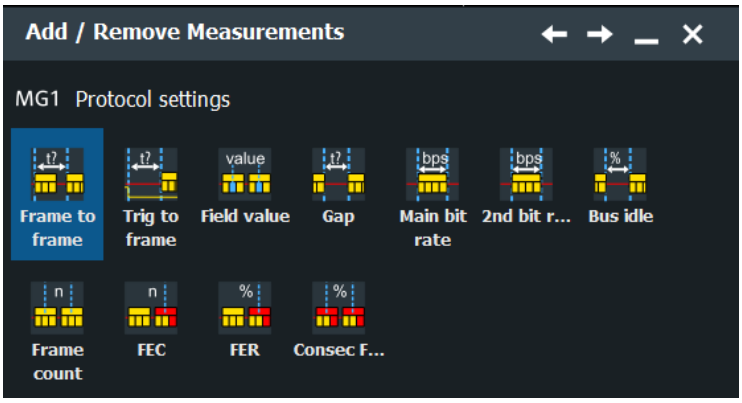


Table 8-10: Protocol measurements

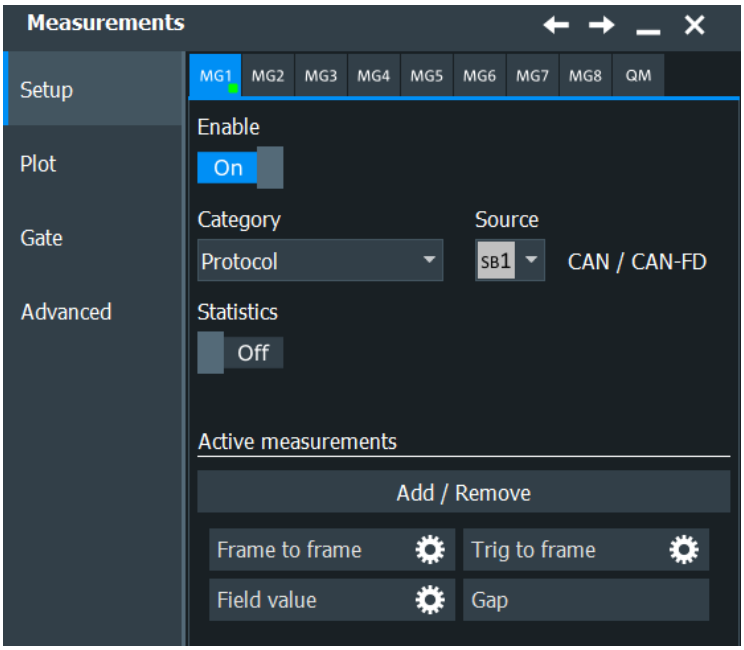
Meas. type	Description / result
Field value	Value of a field over time
Frame to frame	The distance between two frames
Trigger to frame	The distance between a defined frame and the next trigger signal
Gap	Measures a gap, period at which the bus is idle. The distance of a gap can only be measured between two identified frames.
Main bit rate	Bit rate as defined by the standard
2nd bit rate	Additional bit rate, protocol dependent For example: "Data rate" for the CAN-FD protocol.
Bus idle	Calculates the bus idle time
Frame count	Number of all frames within the acquisition window
Frame error count	Sum of all frames with errors within the acquisition window

Meas. type	Description / result
Frame error rate	Sum of all frames with errors divided by all frames within the acquisition window
Consecutive frame error rate	Measures the rate at which at least two consecutive frames have an error

8.2.11.2 Settings for protocol measurements

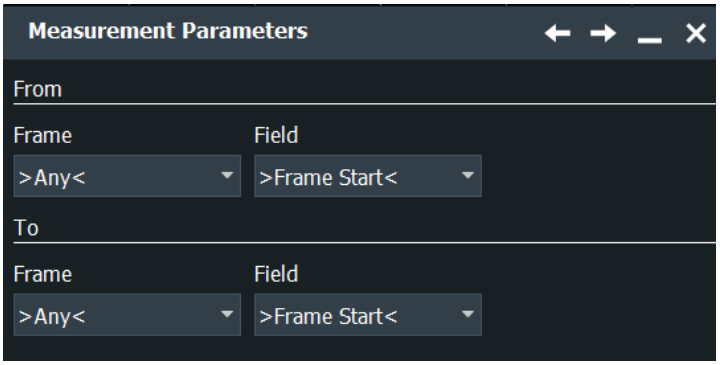
Access: [Measure] > "Setup" tab > "Protocol" category

Protocol measurements require that a supported protocol is enabled first.



If further settings are available for a measurement, a settings icon is shown beside the measurement's name in the "Active Measurements" list. Tap the icon to configure the measurement.

Frame to frame
Opens a dialog to set up the parameters for the frame to frame measurement. You can define a start frame ("From") and an end frame ("To") for the measurement.



The types of available frames and fields are depending on the enabled protocol.

Table 8-11: Available protocol frames

Protocol	Frame	Field
I2C	W R	Address
		2. Addr
		ACK
		R/W
		Data
SPI	MISO MOSI	Word
CAN/CAN FD	Remote Data	ID FDF BRS ESI DLC SCV P SC Data CRC ACK
	Overload	-
LIN	WAKE	-
	Data	ID Data

Protocol	Frame	Field
RFFE	Register 0 Write Register Write Register Read Extended Register Write Extended Register Read Extended Register Write Long Extended Register Read Long Interrupt Summary and Identification Masked Write Master Ownership Handover Master Write Master Read Master Context Transfer Write Master Context Transfer Read	SA MID BC Addr Mask Data
Ethernet	MAC Sleep EOS	Preamble SFD Dest Src Len/Type Data FCS
	Idle	Triplet
100BASE-T1	MAC Fill	Preamble SFD Dest Src Len/Type Data FCS
	Idle	Triplet

Also, if a label list is loaded and enabled for the protocol, the symbolic label names can be selected from the "Frame"/"Field" value list.

"Frame" Selects the type of start frame (from) / end frame (to).

"Field" Selects the type of start field (from) / end field (to).

"Value" Sets a value for the selected field.

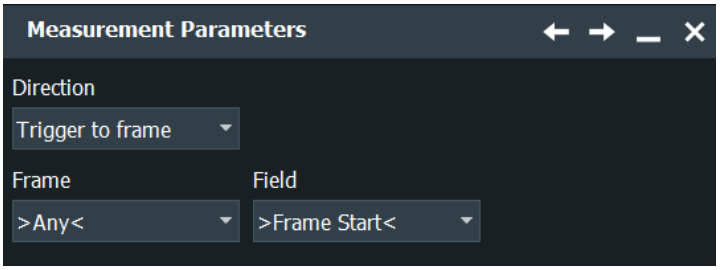
Remote command:

`MEASurement<m>:PROTocol:F2FRame:FRMFrom` on page 1541
`MEASurement<m>:PROTocol:F2FRame:FRMTo` on page 1541
`MEASurement<m>:PROTocol:F2FRame:FLDFrom` on page 1540
`MEASurement<m>:PROTocol:F2FRame:FLDTo` on page 1540

MEASurement<m>:PROTocol:F2FRame:VALFrom on page 1541
MEASurement<m>:PROTocol:F2FRame:VALTo on page 1541

Trigger to frame
Opens a dialog to set up the parameters for the trigger to frame measurement.

The types of available frames are depending on the enabled protocol, see Table 8-11.



"Direction" Selects the direction for the measurement, from the trigger to the next frame (start),

"Frame" Selects the type of frame.

"Field" Selects the type of field.

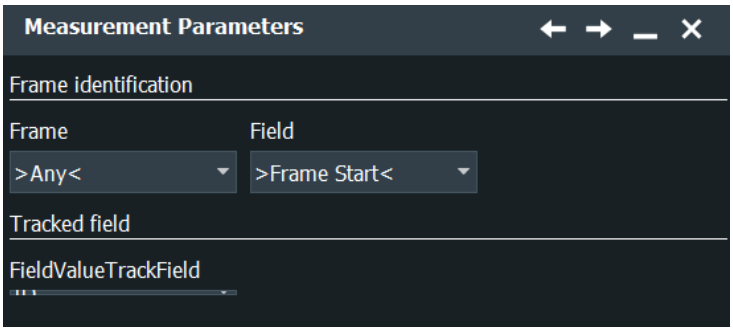
"Value" Sets a value for the selected field.

Remote command:
MEASurement<m>:PROTocol:T2FRame:DIRection on page 1544
MEASurement<m>:PROTocol:T2FRame:FLD on page 1544
MEASurement<m>:PROTocol:T2FRame:FRM on page 1544
MEASurement<m>:PROTocol:T2FRame:VALue on page 1545

Field value
Opens a dialog to set up the parameters for the field value measurement. You can first identify a frame with specific field value and then track a selected field of this frame.

The types of available frames and fields depend on the enabled protocol, see Table 8-11.

If a label list is loaded and enabled for the protocol, sometimes there are defined calculations and formatting for the displayed field value. In this case, the measurement displays the value according to the definition in the label list.



"Frame" Selects the type of frame.

"Field" Selects the type of field.

"Value" Sets a value for the selected field.

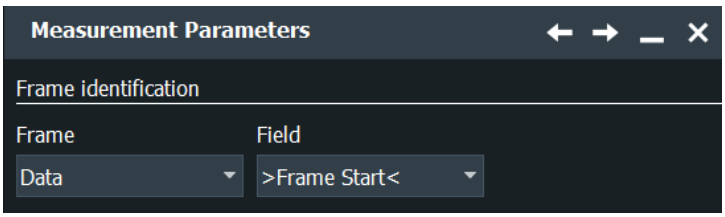
"Field to be tracked" Selects the type of field which value is tracked over time.

Remote command:
`MEASurement<m>:PROTocol:FLDValue:FLD` on page 1542
`MEASurement<m>:PROTocol:FLDValue:FRM` on page 1542
`MEASurement<m>:PROTocol:FLDValue:TRCK` on page 1542
`MEASurement<m>:PROTocol:FLDValue:VAL` on page 1542

Main bit rate / 2nd bit rate
Opens a dialog to set up the parameters for the main bit rate/ 2nd bit rate measurement. You can identify a frame and a specific field value.

The types of available frames and fields depend on the enabled protocol, see Table 8-11.

If a label list is loaded and enabled for the protocol, sometimes there are defined calculations and formatting for the displayed field value. In this case, the measurement displays the value according to the definition in the label list.



"Frame" Selects the type of frame.

"Field" Selects the type of field.

"Value" Sets a value for the selected field.

Remote command:
`MEASurement<m>:PROTocol:MBITrate:FLD` on page 1543
`MEASurement<m>:PROTocol:MBITrate:FRM` on page 1543
`MEASurement<m>:PROTocol:MBITrate:VAL` on page 1543
`MEASurement<m>:PROTocol:SBITrate:FLD` on page 1543
`MEASurement<m>:PROTocol:SBITrate:FRM` on page 1543
`MEASurement<m>:PROTocol:SBITrate:VAL` on page 1544

8.2.12 Result analysis

The behavior of measurement results over time can be evaluated in different ways:

- Statistics
- Long-term measurements
- Histograms on measurement results
- Track

8.2.12.1 Statistics

Statistics can be compiled for each measurement group separately, and also for long-term measurements. If enabled, statistical results are shown in the result box.

By default, all events in each acquisition are measured and incuded in the statitics: the measurement result is not only determined once within one acquisition, but repeatedly, if available. More results provide a larger basis for statistical evaluation.

To obtain precise results, additional measurement settings can be useful:

- Reference/signal levels: configuring user-defined levels can compensate for irregular data, see Chapter 8.2.4, "Reference levels", on page 307.
- Gate areas: restricting the waveform range for measurement can eliminate irregular data, see Chapter 8.2.3, "Measurement gates", on page 304.
- Defining a "Signal threshold" for time, area and counting measurements can eliminate noise from the evaluation, see "Signal threshold" on page 322.
- Spectrum measurements: you can eliminate noise from the evaluation, see Threshold and "Peak excursion" on page 295

► To enable statistics, use one of the following ways.
Make sure to select the subtab of the measurement group for which you want to compile statistics.

- On the toolbar assist, enable "Statistics".
- On the "Setup" tab, enable "Statistics".

If statistics are enabled, the following results are calculated. You can select which results are shown in the result table, see Chapter 8.2.2.3, "Display settings for results", on page 303.

Label	Description
+Peak	Positive peak value (maximum)
-Peak	Negative peak value (minimum)
µ (Avg)	Average
RMS	Root mean square
σ (S-dev)	Standard deviation
Event count	Number of measured events (e.g. rising edges, pulses etc.)
Wave count	Number of waveforms (acquisitions) the measurement is based on

Remote commands:

- `MEASurement<m>:RESult[:ACTual]?` on page 1487
- `MEASurement<m>:RESult:AVG?` on page 1487
- `MEASurement<m>:RESult:COUNt?` on page 1489
- `MEASurement<m>:RESult:EVTCount?` on page 1487
- `MEASurement<m>:RESult:NPEak?` on page 1487

- MEASurement<m>:RESult:PPEak? on page 1487
- MEASurement<m>:RESult:RMS? on page 1487
- MEASurement<m>:RESult:STDDev? on page 1487
- MEASurement<m>:RESult:WFMCount? on page 1487
- MEASurement<m>:RESult:STARt? on page 1489
- MEASurement<m>:RESult:STOP? on page 1489
- MEASurement<m>:ARNames on page 1486
- MEASurement<m>:ARES? on page 1486

The peak and average values and the standard deviation of the long-term points are also shown in the graph of the long-term measurement.

Stopping and starting the acquisition does not reset statistics but only stops and continues them.

The instrument only resets statistical evaluation if you change measurement setup:

- Select measurement
- Create or modify gate
- Enable/disable long-term measurement and histogram
- Enable continuous autoscale with enabled histogram
- Switch on/off channels
- Enable/disable cursors
- Tap "Reset" or "Clear sreen results"

After a reset, new statistics are compiled beginning with the next acquired waveform.

If limit and margin check is enabled, the icons in the result table indicate if statistical results exceed a limit or margin. These violations do not initiate an action.

8.2.12.2 Long-term measurements

Long-term measurements show the behavior of measurement results over a longer time or for many samples. Therefore the measurement results of a specified time period are summarized into one long-term point. For each point, the current value measured at the end of the time period is written to the long-term waveform. In addition, statistical results for each time period are calculated, saved, and reset. This reset avoids constantly rising maximum or constantly falling minimum values until the end of the measurement.

You can define the number of long-term points and export the long-term data, including statistical results. The measurement histogram is a vertical histogram shown in the long-term diagram.

See also: "Enable (Long term)" on page 360.

Performing long-term measurements

1. On the "Meas" menu, select "Result Analysis".

2. Select the subtab for the measurement group you want to configure.
3. Under "Long term", tap "Enable".
4. Since the waveform can change in the process of time, enable "Continuous auto scale" to adapt the scaling automatically.
   Alternatively, tap the "Auto scale" to adjust the scale once and to see the long-term waveform.
5. Tap "Horizontal scaling".
6. Define the "Number of points" to be shown in the long-term diagram.
7. Set the "Scale mode" that defines the period of time from which one long-term point is created.
   See "Scale mode" on page 362 for setting details.

   For each long-term measurement point, the current measurement value is added to the long-term waveform.
8. If you need the statistical data of the long-term points:
   a) Tap "Result Analysis" to return to the measurement settings.
   b) Enable statistics.
   c) Let the measurement run and export the data when finished.

8.2.12.3 Histograms on measurement results

Histograms are available not only for channel, math and reference waveforms, but also on measurement results. These histograms cannot be configured, and they are shown in a separate diagram. The source is a measurement, and all events in all acquisitions are are measured and included in the histogram. If the histogram is based on long-term measurement, it is shown in the long-term diagram.

See also: "Enable (Histogram)" on page 360.

8.2.12.4 Track

The track is a waveform that shows measurement values in time-correlation to the measured signal. It is the graphical interpretation of all measurement values of a single acquisition.

The track is available for most amplitude/time measurements (except for High, Low, Amplitude, Max, Min, Peak to peak, Mean, RMS, S-dev, Pos. and Neg. overshoot, and Area), and for jitter measurements.

Enabling the track enables also the Continuous auto scale and Measure all events in each acquisition.

To analyze the track, you can use cursor measurements and zoom.

Creating a track waveform

1. Configure the measurement group.

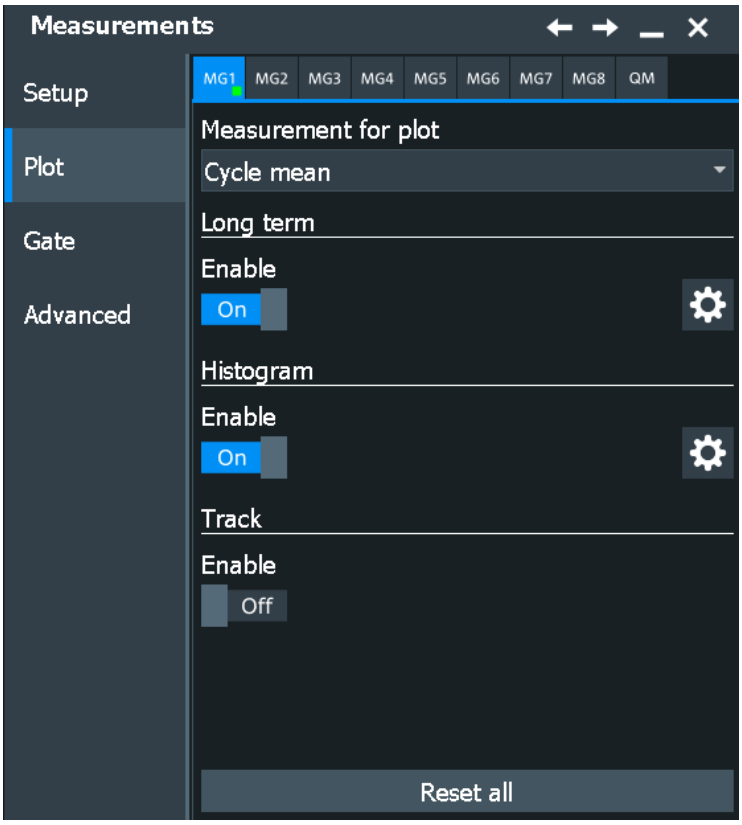
2. Activate the measurement group.
3. Select the "Result Analysis" tab.
4. Enable the track.

See also: "Enable (Track)" on page 361.

8.2.12.5 Settings for result analysis

The settings in the "Result Analysis" tab activate and configure long-term measurements, statistical calculations, and the histogram of measurement results over a longer period of time.

For scaling settings of the long-term diagram, see Chapter 8.2.12.6, "Horizontal long-term scaling", on page 362.



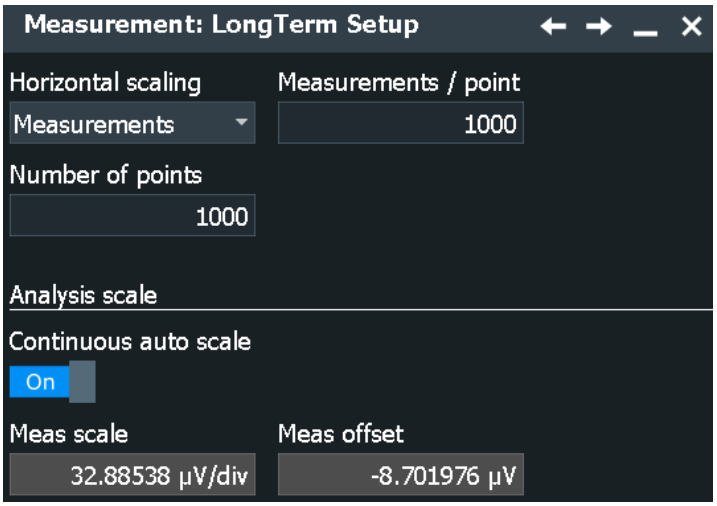
Measurement for plot
Selects the measurement that is used as source for mathematic calculations, long-term measurements, and histograms. These evaluations are performed on one measurement only, not on all measurements of the group. Statistics are calculated for all measurements.

The selected measurement cannot be disabled in the "Add / Remove Measurement" dialog box.

Remote command:
`MEASurement<m>:MAIN` on page 1483

Enable (Long term)
Enables long-term measurement of the main measurement.

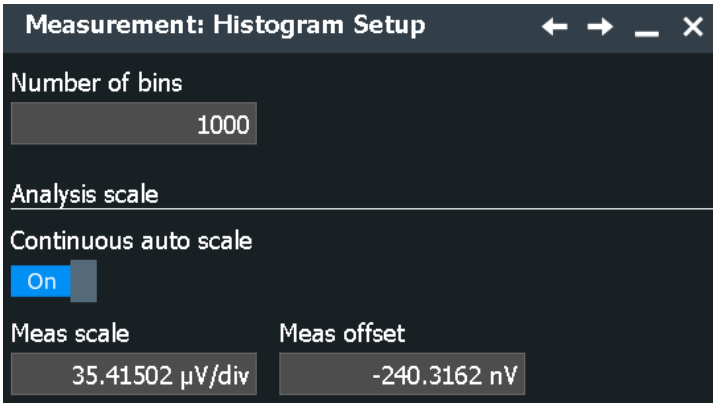
Long-term measurements are performed on the "Measurement for plot".



Remote command:
`MEASurement<m>:LTMeas[:STATe]` on page 1538

Enable (Histogram)
Displays a histogram of measurement results - the cumulative occurrence distribution of measurement results in a graphic. Enabling the histogram enables also the statistics and "Measure all events in each acquisition".

Long-term measurements are performed on the "Measurement for plot".



Remote command:
`MEASurement<m>:STATistics:HISTogram` on page 1534

Number of bins
Sets the number of bins - the number of vertical bars that build the histogram.

If "Continuous auto scale" is enabled, the instrument determines the number of bins based on the timebase, the current measurements, and other settings. To set the number of bins manually, disable "Continuous auto scale".

Remote command:
`MEASurement<m>:STATistics:HBINs` on page 1535

Enable (Track)
Enables the track of measurement results over time and displays the track waveform. It is the graphical interpretation of all measurement values of a single acquisition.

The track is available for most amplitude/time measurements (except for High, Low, Amplitude, Max, Min, Peak to peak, Mean, RMS, S-dev, Pos. and Neg. overshoot, and Area), and for jitter measurements.

Enabling the track enables also the Continuous auto scale and Measure all events in each acquisition.

Before you can enable the track, activate the appropriate measurement.

Remote command:
`MEASurement<m>:TRACk[:STATe]` on page 1545

Reset
Immediately resets the histogram, the long-term measurement and the statistics.

Stopping and starting the acquisition does not reset these analyses but only stops and continues them.

To delete all results, waveforms and history, select "Display" menu > "Clear all".

Remote command:
`MEASurement<m>:STATistics:RESet` on page 1536

Analysis scale
The measurement scale of a long-term measurement diagram or measurement histogram can be set automatically by the instrument, or manually.

Use automatic scaling if the measurement is running and you cannot see the expected results.

"Continuous auto scale"
Performs an automatic scaling whenever the long-term waveform or the histogram does not fit in the diagram during the measurement period.

"Vertical"
Defines the vertical scaling per division for long-term measurement period and the measurement histogram.

"Offset"
Defines an offset for the long-term measurement and the measurement histogram.

Remote command:
`MEASurement<m>:VERTical:CONT` on page 1537
`MEASurement<m>:VERTical:AUTO` on page 1537
`MEASurement<m>:VERTical:SCALe` on page 1538
`MEASurement<m>:VERTical:OFFSet` on page 1537

Measure all events in each acquisition
Normally, only one measurement is performed for each acquired waveform to get best performance. If "Measure all events in each acquisition" is enabled, more than one result is taken from one acquired waveform and the results are included evaluation. For example, the rise time is measured on all pulses in the waveform, not only on the first.

The result box shows only the first result of the waveform, the following results are used only for evaluation.

All event results are also considered in limit and margin checks and can initiate an action. However, the icons in the result box only indicate violations of the first result.

Measuring all events is enabled automatically when calculating statistics and histograms. It can also be useful when generating tracks; however, it reduces the performance of the instrument.

The number of considered results can be restricted: "Limit" sets the maximum number of measurement results per acquisition.

Remote command:
`MEASurement<m>:MULTiple` on page 1534
`MEASurement<m>:MNOMeas` on page 1534

8.2.12.6 Horizontal long-term scaling

In this dialog box, you define the horizontal scale of long-term measurement diagrams. The length of the long-term measurement is defined by the number of points.

If option R&S RTO6-K510 I²S Audio Signals is installed, the trend diagram is configured here.

Number of points
Defines the total number of points to be displayed in the long-term measurement diagram.

Remote command:
`MEASurement<m>:LTMeas:COUNt` on page 1538

Scale mode
Defines when the points of a long-term measurement are created.

If statistics are enabled, each long-term measurement point shows the statistical mean and standard deviation of the results measured during the defined period.

If statistics are disabled, the first measurement result of each period is taken as long-term measurement point.

"Time"	Sets one long-term measurement point for the time defined in "Time/point".
"Waveforms"	Sets one long-term measurement point for several acquired waveforms. The number is defined in "Waveforms/point".
"Measurements"	Sets one long-term measurement point for several measurement results. The number is defined in "Measurements/point".

Remote command:
`MEASurement<m>:STATistics:MODE` on page 1535

Time / point
Defines the time to create one point of the long-term measurement. The "Time / point" value is a lower time limit. The actual time between two points depends on the acquisition and postprocessing time.

The long-term measurement is not a data logger with equidistant points as the time between two points varies.

This setting is only available if "Scale mode" is set to "Time".

Remote command:
`MEASurement<m>:STATistics:RTIMe` on page 1536

Measurement time
Defines the total duration of the long-term measurement: *Time/point * Number of points*.

This setting is only available if "Scale mode" is set to "Time".

Remote command:
`MEASurement<m>:LTMeas:TIME` on page 1539

Wfms / point
Defines the number of measured waveforms from which one point of the long-term measurement is created.

This setting is only available if "Scale mode" is set to "Waveforms".

Remote command:
`MEASurement<m>:STATistics:RCOunt` on page 1535

Measurements / point
Defines the number of measurement results from which one point of the long-term measurement is created.

This setting is only available if "Scale mode" is set to "Measurements".

Remote command:
`MEASurement<m>:STATistics:RMEascount` on page 1536

8.2.13 Limit and margin checks

Limit and margin checks evaluate whether the measurement result exceeds a specified value. Violations are indicated by icons in the result box. Furthermore, you can define actions that are performed on limit or margin violation, like saving the waveform or measurement results.

The following results are considered in limit and margin checks:

- All selected measurements.
- All measured events in an acquisition: all results can initiate an action. However, the icons in the result box indicate only violations of the first result.

- Statistical results. Limit and margin violations of statistical results are indicated by icons in the result box. These violations do not initiate an action.

See also: Chapter 8.2.2.1, "Measurement status", on page 302

8.2.13.1 Performing limit checks

1. Open the menu and select "Measure".
2. Select the "Advanced" tab.
3. Select the subtab of the measurement group that you want to configure.
4. Under "Type", select "Limit only" to distinguish only between valid and non-valid values.
   Select "Margin&Limit" to check two values, where the margin is still valid, while the limit is not.
5. Tap "Setup". Define the valid value range for each active measurement.
   The margins must always be within the valid value range. If necessary, the limit or margin values are adapted to match the selected valid range.
   See also "Upper limit, Lower limit, Upper margin, Lower margin, Valid range" on page 365
6. Define what happens when the defined limits and margins are exceeded.

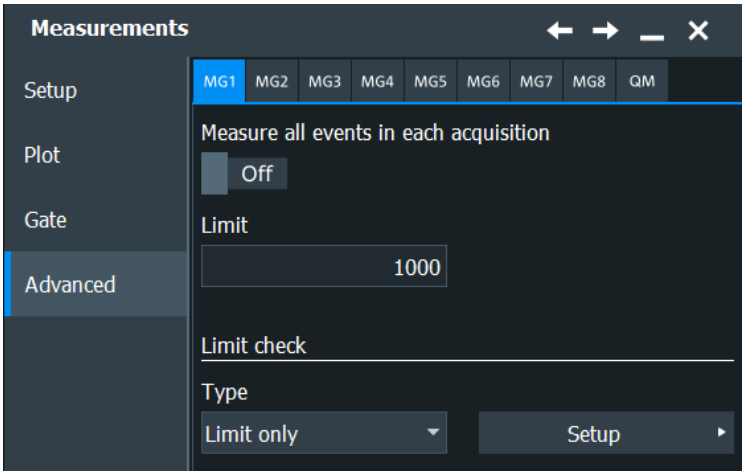
   For each action, define when the instrument starts it:
   - If the limits or margins are exceeded
   - If the measurement is completed without limit violations
   - Not at all

   As a result of the limit check, the specified actions are performed and an icon indicates the status in the result box.

8.2.13.2 Limit and margin settings

Access: [Measure] > "Advanced" tab

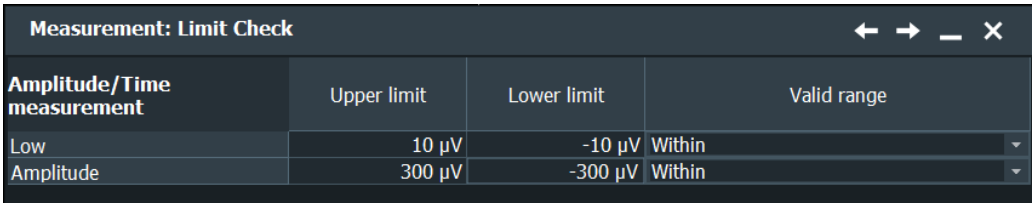
If a check is selected, tap "Setup" to set the limit and margin values and the range of valid measurement results.



Type
Selects and enables the limit or margin check.

"Off"	No limit check is performed.
"Limit only"	Limits are checked for violation.
"Margin & Limit"	Margins and limits are checked for violation.

Upper limit, Lower limit, Upper margin, Lower margin, Valid range
Set the limits and margins for each measurement, and also specify the valid range.



Amplitude/Time measurement	Upper limit	Lower limit	Valid range
Low	10 µV	-10 µV	Within
Amplitude	300 µV	-300 µV	Within

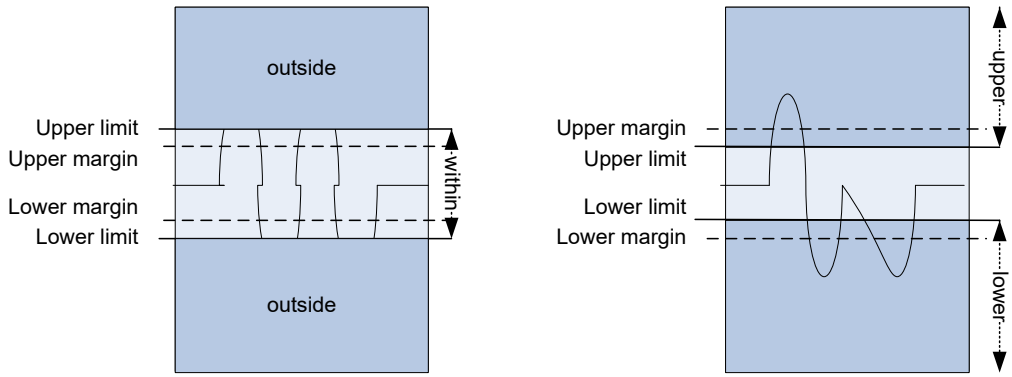


Figure 8-15: Limit and margin definition

Limits are stricter than the margins for the value check. Thus, the margins must be within the valid range. If necessary, the limit and margin values are adapted according to the selected valid range.

The settings are only visible if "Limit check" is enabled.

Remote command:
`MEASurement<m>:AMPTime:LCHeck<n>:LOWer:LIMit` on page 1499
`MEASurement<m>:AMPTime:LCHeck<n>:LOWer:MARGin` on page 1499
`MEASurement<m>:AMPTime:LCHeck<n>:UPPer:LIMit` on page 1499
`MEASurement<m>:AMPTime:LCHeck<n>:UPPer:MARGin` on page 1499
`MEASurement<m>:AMPTime:LCHeck<n>:VALid` on page 1498
To check limits and margins of jitter measurements, use the AMPTime remote commands.
`MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:LIMit` on page 1510
`MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:MARGin` on page 1511
`MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:LIMit` on page 1510
`MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:MARGin` on page 1511
`MEASurement<m>:EYEJitter:LCHeck<n>:VALid` on page 1510
`MEASurement<m>:HISTogram:LCHeck<n>:LOWer:LIMit` on page 1530
`MEASurement<m>:HISTogram:LCHeck<n>:LOWer:MARGin` on page 1531
`MEASurement<m>:HISTogram:LCHeck<n>:UPPer:LIMit` on page 1530
`MEASurement<m>:HISTogram:LCHeck<n>:UPPer:MARGin` on page 1531
`MEASurement<m>:HISTogram:LCHeck<n>:VALid` on page 1530
`MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:LIMit` on page 1517
`MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:MARGin` on page 1518
`MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:LIMit` on page 1517
`MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:MARGin` on page 1518
`MEASurement<m>:SPECtrum:LCHeck<n>:VALid` on page 1517

8.2.13.3 Actions on limit check results

To define what happens when the limits and margins are exceeded, tap Action on limit event below the limit and margin ranges.

Actions are initiated by all measurements and all results measured on one acquisition.

Note that the violation actions do not distinguish between a margin violation and a limit violation. However, different icons are displayed in the result box.

For each action, you can define the event on which the action is initiated:

- On violation
  The action is initiated when the limits or margins are exceeded during the measurement.
- On successful completion
  The action is initiated when a defined number of acquisitions has been captured, and the limits or margins were not exceeded.

Independent of these actions, an icon is displayed in the result box, see Chapter 8.2.2, "Measurement results", on page 301.

Beep
Generates a beep sound.

Remote command:
`MEASurement<m>:ONViolation:BEEP` on page 1550

Stop acq
Stops data acquisition on violation.

Remote command:
`MEASurement<m>:ONViolation:ACQStop` on page 1550

Save Wfm
Saves the waveform data to the file.

Remote command:
`MEASurement<m>:ONViolation:WFMSave` on page 1551

Trigger Out Pulse
Creates a pulse on the Trigger Out connector on limit violation.

When "Trigger Out Pulse" is used, the trigger control option "Enable trigger out" is disabled. Thus, the trigger-out pulse is created only on limit violation but not when a trigger occurs. The pulse is provided always with the minimum delay of 800 ns, the "Delay" cannot be set.

See also: "Trigger out signal setup" on page 225.

Remote command:
`MEASurement<m>:ONViolation:TRIGgerout` on page 1551

Report
Creates and saves a report using the settings defined in "File" menu > "Report Setup".

Remote command:
`MEASurement<m>:ONViolation:REPort` on page 1551

Start Executable
Starts an external application. Tap "Config Executable" to set the application path and parameters.

Remote command:
`MEASurement<m>:ONViolation:RUNexec` on page 1551

8.3 Quick measurements

Quick measurement performs a set of up to eight amplitude/time measurements on one source, simply by tapping the "Quick measurement" toolbar icon. The results are displayed in a results box. You can configure the measurement to be included in quick measurement. The current configuration can be saved to repeat the measurement quickly.

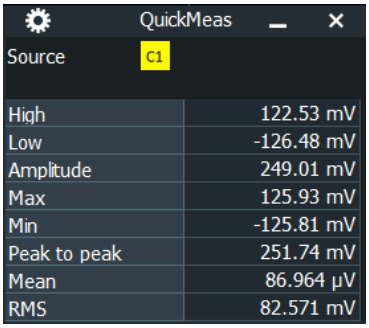
8.3.1 Starting the quick measurement

If the "Quick meas" icon is not visible on the toolbar, add it to the toolbar: see Chapter 3.3.6.2, "Configuring the toolbar", on page 53.

1. Tap the waveform that you want to measure.
2. Tap the "Quick measurement" icon on the toolbar.

3. Tap the diagram.

   The result box shows the results of the default quick measurement.



8.3.2 Configuring the quick measurement

The default configuration of the quick measurement includes already 8 amplitude measurements. If these measurements do not fit the measurement task, you can modify the selection.

1. Open "Menu" > "Measure" > "Setup".
2. Select the "QM" subtab.

3. Tap "Add/Remove Measurements".
4. Disable all measurements that you do not need.

5. Select the measurements that you want to add to the quick measurement.
6. Tap "OK".
7. A "Settings" icon indicates whether further settings are required. Tap the measurement. A dialog box opens, where you can adjust the measurement.
8. Tap "Set as QuickMeas" to save the setup.

   The saved iQuickMeas setup is used when you start a new quick measurement.
9. "Enable" the measurement.

Set as QuickMeas
Saves the current QuickMeas setup. The saved QuickMeas configuration is used when you start a new quick measurement. It remains until you save another QuickMeas setup, or until you reset the instrument to "Factory defaults". The "Set as QuickMeas" is not available if the current configuration already has been saved.

9 Spectrum analysis

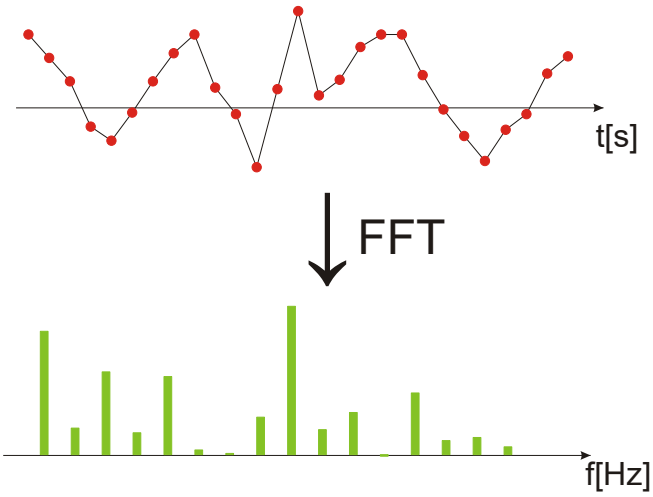
The R&S RTO6 provides two ways of spectrum analysis:

- Basic FFT calculation, which is included in the firmware
- Spectrum analysis option R&S RTO6-K37, which provides a wide range of analysis possibilities, for example, spectrogram, cursor and automatic measurements.

9.1 FFT analysis

9.1.1 Fundamentals of FFT analysis

During FFT analysis, a signal in the time domain is converted to a spectrum of frequencies. As a result, either the magnitude or the phase of the determined frequencies can be displayed. FFT analysis can be restricted to an extract of the original time base, and the results display can be restricted to a specified frequency range.

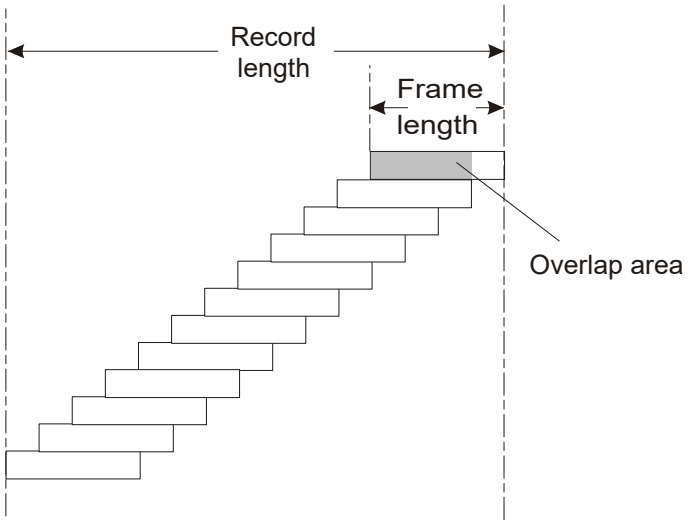


Frames/Segments

To convert the time domain signal to a frequency spectrum, an FFT (Fast Fourier Transformation) unit is used which converts a vector of input values into a discrete spectrum of frequencies.

Conventional oscilloscopes calculate one FFT per capture. The R&S RTO6 can calculate multiple FFTs per capture by dividing one capture into several *segments*, or *frames*. Thus, the R&S RTO6 can visualize how the frequency content of a signal changes over time which helps to detect intermittent or sporadic signal details. Furthermore, the R&S RTO6 allows consecutive frames to overlap. This is especially useful in conjunction with window functions since it enables a gap-free frequency analysis of the signal.

The overlapping factor can be set freely. The higher the overlap factor, the more frames are used. This leads to more individual results and improves detection of transient signal effects. However, it also extends the duration of the calculation. The size of the frame depends on the number of input signal values (record length), the overlap factor, and the FFT size (number of samples used for FFT calculation).



Window functions

Each frame is multiplied with a specific window function after sampling in the time domain. Windowing helps minimize the discontinuities at the end of the measured signal interval and thus reduces the effect of spectral leakage, increasing the frequency resolution.

There are several window functions that can be used in FFT analysis. Each of the window functions has specific characteristics, including some advantages and some tradeoffs. Consider these characteristics carefully to find the optimum solution for the measurement task.

For details, see "Window type" on page 381.

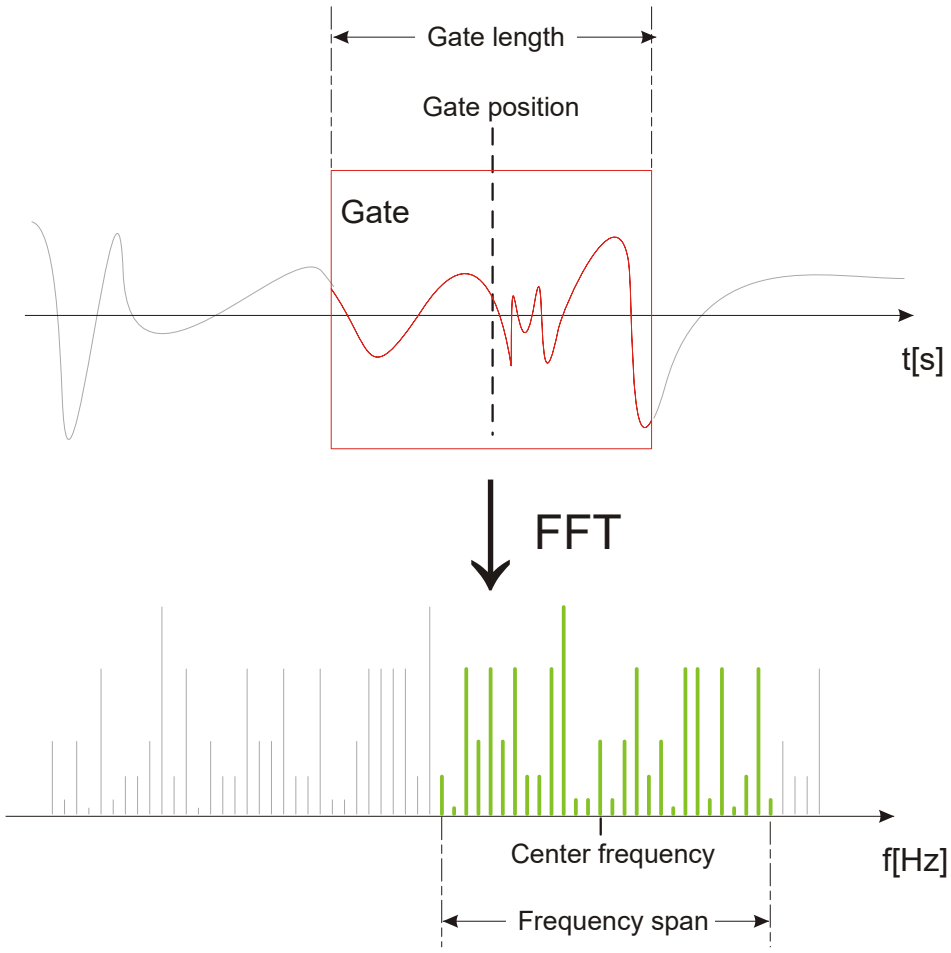
Combining FFT results

After the window function has been applied to the FFT results for each segment, the results for all segments of the data acquisition must be combined to obtain the final waveform. Various arithmetic functions are available for FFT segments, such as averaging, enveloping, or minimum and maximum calculation.

Gating functions

You can restrict the time base of the input signal for which FFT analysis is to be performed. There are various methods to do so:

- Define absolute start and stop times for the time base extract
- Define relative start and stop values that define a percentage of the original time base
- Couple the time base extract for FFT to an active zoom area.



Restricting the result range

You can restrict the results of the FFT analysis to a specified frequency range. The frequency range can be defined in two ways:

- Define a center frequency and frequency span
- Define start and stop frequencies

Using the new cursor functions for spectrum waveforms you can easily determine the results for the current center frequency by moving the cursor to that frequency ("C1 to Center"). If you detect a point of interest in the spectrum diagram, you can place the cursor on it and then move the center frequency to the position of the cursor automatically ("Center to C1").

See Chapter 8.1, "Cursor measurements", on page 284.

Magnitude vs. phase display

The result of an FFT analysis is a spectrum of frequencies. Either the magnitudes or the phases of those frequencies are displayed, depending on the used FFT function. In "Basic" mode, and for the "Advanced" mode FFT functions |FFT|, FFT (re) and FFT

(im), the magnitude is displayed. For the "Advanced" mode FFT (φ) function, the phase is displayed.

For magnitude display, you can select the scale and range of magnitudes to be displayed. For linear scaling, the vertical value range of the input signal is used. For logarithmic scaling, the logarithmic power of the frequency is displayed. In this case, the input signal must be given in either Volt or Watt. The resulting value range is defined by a maximum value and a range size. Logarithmic scaling can also be set in relation to a given reference value.

For phase display, you can select the unit and suppress phases beneath a threshold value which are most likely caused by noise. The value range [-π, + π] or [-180°, +180°] is used. Phase shifts due to a limitation of the value range can be eliminated using the "Unwrap" function.

Dependencies between FFT parameters

FFT analysis in the R&S RTO6 is highly configurable. Several parameters, including the resolution bandwidth, frequency span and center frequency, can be defined according to your requirements. Note, however, that several parameters are correlated and not all can be configured independently of the others.

The **resolution bandwidth** defines the minimum frequency separation at which the individual components of a spectrum can be distinguished. Small values result in high precision, as the distance between two distinguishable frequencies is small. Higher values decrease the precision, but increase measurement speed.

The minimum achievable RBW depends on the integration time which is equivalent to the number of samples available for FFT calculation. If a higher spectral resolution is required, the number of samples must be increased by using a higher sample rate or longer record length. To simplify operation some parameters are coupled and automatically calculated, such as record length and RBW.

The **frequency span** and **center frequency** define the start and stop frequency of the spectral diagram. By default, a suitable frequency range according to the resolution bandwidth is selected, in respect to performance and precision. Span and RBW settings are coupled, so that the parameters can be adjusted automatically as necessary.

With a **Span/RBW ratio** of 100 and a screen resolution of 1000 pixels, each frequency in the spectrum is displayed by 10 pixels. A span/RBW ratio of 1000 provides the highest resolution. For full flexibility, the span/RBW coupling can also be disabled. Note, however, that a higher span/RBW ratio (i.e. low RBW values and large frequency spans) result in large amounts of data and extend the duration of the calculation.

Advanced FFT functions

In "Advanced" math definition mode, other FFT results than the basic frequency magnitude can be displayed.

- **FFT (φ)**: phase display
- **FFT (im)**: imaginary part of FFT value (magnitude)
- **FFT (re)**: real part of FFT value (magnitude)

- **FFT -dφ*df** (group delay): the negative derivative of the phase with respect to frequency; useful to measure phase distortion

9.1.2 Configuring spectrum waveforms

During FFT analysis, a signal in the time domain is converted to a spectrum of frequencies. A basic spectrum waveform can be displayed quickly. By defining additional FFT parameters, the waveform can be configured in more detail.

As a result, either the magnitude or the phase of the determined frequencies can be displayed, or more complex FFT functions. Analysis can be restricted to an extract of the original time base, and the results display can be restricted to a specified frequency range.

To display a basic spectrum waveform using the toolbar

1. Add the "FFT" icon to the toolbar, see Chapter 3.3.6.2, "Configuring the toolbar", on page 53 .

2. Tap the "FFT" icon on the toolbar.

   The FFT overlay menu opens.

3. Tap the waveform for which the FFT is performed.

   The first available math waveform is configured to use the selected waveform as a source and the "Mag(FFT(x))" operator, and is enabled. The spectrum waveform is displayed in a new diagram.

To display a basic spectrum waveform using the [FFT] key

1. Press the [FFT] key.

   The "FFT" tab of the "Math" dialog opens.

2. Set the "Source" to the input signal.
3. Enable "Display".
4. If necessary, edit the spectrum waveform parameters as described in the following procedures.

To display advanced spectrum waveforms

Using the "Formula Editor", you can display other FFT results than the basic frequency magnitude, and use more FFT sources.

1. Open the "Menu" > "Math" dialog.
2. Select the Equation tab.

3. Double-tap the editing area.

   The "Formula Editor" is displayed.

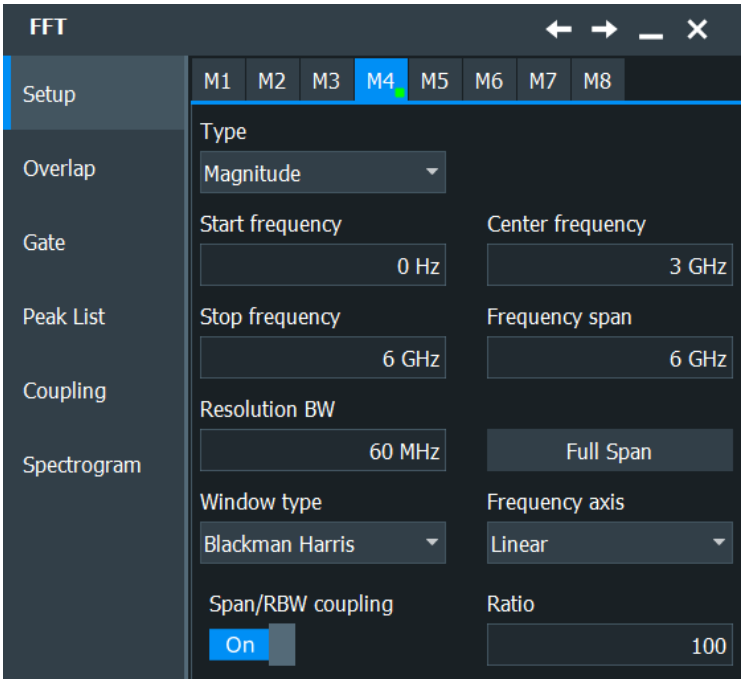
4. Delete the contents of the edit field.
5. Tap the "More" key to display further functions in the editor.
6. Tap the required FFT function key.
7. Select the source channel.
8. Close the parenthesis.
9. Tap "Enter"
10. Select the "Scale" tab to adjust the vertical and horizontal scale of the spectrum.
11. On the "Setup" tab, select "FFT setup" to configure the spectrum. See "To configure the spectrum of FFT analysis" on page 375.

To configure the spectrum of FFT analysis

By default, a suitable frequency range for the expected horizontal values according to the resolution bandwidth is selected, in respect to performance and precision. Span and RBW settings are coupled. If a more precise evaluation is required, for example for postprocessing in a different application, disable the coupling and change the frequency ranges and resolution bandwidth values as required.

1. Open [FFT] > "FFT setup".

   The "FFT" dialog opens.



2. In the "Setup" tab, tap the "Frequency axis" button to select the type of scaling you want to use: linear or logarithmic.

3. Disable the "Span/RBW coupling".
4. Specify the frequency range you want to display using one of the following methods:
   - Enter a "Center frequency" and a "Frequency span" that define the spectrum.
   - Enter a "Start frequency" and "Stop frequency" that define the spectrum.
   - Tap the "Full Span" button to display the complete spectrum resulting from the FFT analysis.
5. Define the resolution bandwidth for the FFT result.
   The resolution bandwidth defines how precise the results are, i.e. how close together the individual frequencies can be. Small values result in high precision, as the distance between two distinguishable frequencies is small. Higher values decrease the precision, but increase performance.

   You can define the RBW manually, or couple it to other FFT settings. Do one of the following:
   - To couple the RBW to the span, enable the "Span/RBW coupling" option and define the "Ratio". The smaller the ratio, the higher the RBW becomes to display the same frequency span.
   - Enter the "Resolution BW" manually. The "Span/RBW coupling" option is automatically disabled.
   - To couple the RBW to the specified record length, in the " Gate" tab of the "Advanced FFT" dialog box, select the "Record length controlled" option. This option is only available if no gate is being used ("Use Gate" disabled).
6. Select the most suitable "Window type" for your source data. Window functions are multiplied with the input values and thus can improve the FFT display. For details, see "Window type" on page 381.
7. Optionally, select an arithmetic mode for the FFT segments. This mode defines how the individual segment results are combined to a final spectrum waveform. In the "Overlap" tab of the "Advanced FFT" dialog box, tap "Segment arithmetic" and select the required mode from the list.
8. If you use an arithmetic mode, increase the "Overlap factor" for neighboring segments to increase the accuracy of the results.

To restrict the input values (gating)

By default, the FFT is calculated for the entire record length as defined for the data acquisition. However, you can restrict the time range for which the FFT is calculated, resulting in a restricted spectrum. Alternatively, the record length can be determined automatically according to the selected RBW.

1. Open [FFT] > "FFT setup" > "Gate" tab.
2. Determine how the input length is configured by selecting one of the following options:
   - To ensure that the FFT is calculated for the full defined record length, select the "Record length controlled" option. This option is only available if no gate is

being used ("Use Gate" disabled). The RBW is adapted so that the record length can be acquired in the specified acquisition time. However, the RBW is restricted, so that data acquisition may fail if the record length is too long for the specified acquisition time.

- To couple the used record length to the required RBW, select the "RBW controlled" option. This option is only available if no gate is being used ("Use Gate" disabled).
  The required acquisition time for the defined RBW value is indicated.
- To restrict the basis of the FFT calculation to a certain time base, configure a time gate, that is: an extract of the time base in the original diagram. To do so, enable the "Use Gate" option, then do one of the following:
  – Select the "Absolute" mode and enter the "Start" and "Stop" times that define the gate area.
  – Select the "Relative" mode and enter the percentages of the total time base that define the "Relative Start" and "Relative Stop" times.
  – If a zoom area has already been defined in the original diagram and you want to use the same time base for FFT analysis, select "Zoom coupling".

The spectrum waveform displays the spectrum for the specified time span.

To configure magnitude results

1. Open [FFT] > "FFT setup" > "Setup" tab.
2. Set "Type" = "Magnitude".
3. In the "Scale" tab of the "Math" dialog, select the "Unit".
4. Select the "Vertical scale mode" to configure the value range manually or use the automatic settings.
5. For "Vertical scale mode" > "Manual", define the size of the "Vertical range" and the "Vertical maximum" to be displayed.
   For "Vertical scale mode" > "Auto", define the size of the "Vertical range" to be displayed.
   For logarithmic scaling ("Unit = dB"), also define the "Reference level" to be used.

To configure phase results

1. Open [FFT] > "FFT setup" > "Setup" tab.
2. Set "Type" = "Phase".
3. In the "Scale" tab of the "Math" dialog, select the "Unit".
4. To eliminate phase shifts due to a limitation of the value range, enable the "Unwrap" function.
5. To suppress small phase values due to noise, enable the "Suppression" function and enter a "Threshold" value.

To couple spectrum displays

The settings for one or more spectrums can be coupled. Thus, if any FFT setting for any of the coupled spectrums are changed, they are changed for all coupled spectrums.

1. Open [FFT] > "FFT setup" > "Overlap" tab.
2. Select the spectrums that you need to couple. You cannot select the spectrum for the currently selected math waveform. Its settings are applied to the selected spectrums.
3. If necessary, define an FFT function to be used for the coupled math waveforms so that a spectrum is displayed. See Chapter 7.3.1, "Displaying math waveforms", on page 257.

9.1.3 FFT configuration settings

- FFT setup....378
- FFT overlap....382
- FFT gating....384
- Peak list....386
- FFT coupling....386

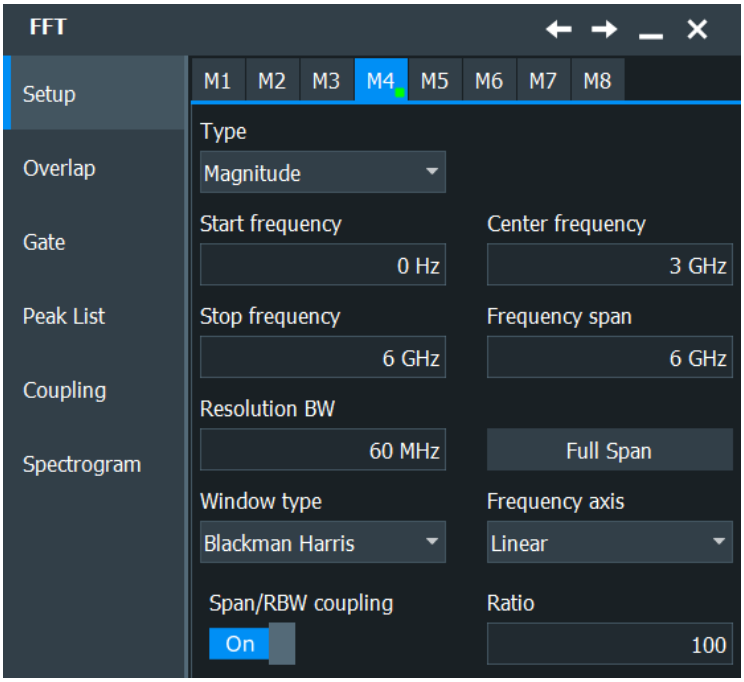
9.1.3.1 FFT setup

Access: [FFT] > "FFT setup" > "Setup" tab

In this tab, you define the settings for the FFT window. The display can be restricted to the results for a certain time base extract and to a specified frequency range.

Additional settings are available on this tab if the Spectrum analysis option (R&S RTO6-K37) is installed.

See Chapter 9.2.4, "Spectrogram configuration settings", on page 392.



Type 379
Display 379
Frequency axis (R&S RTO6-K37 only) 379
Center frequency 380
Frequency span 380
Full span 380
Start frequency 380
Stop frequency 380
Span/RBW Coupling 380
Span/RBW Ratio 381
Resolution BW 381
Window type 381
Use color table 381

Type

Selects between the magnitude and phase spectrum type.

Remote command:

`CALCulate:MATH<m>:FFT:TYPE` on page 1568

Display

If activated, a diagram for the defined math waveform is displayed on the touch screen.

Remote command:

`CALCulate:MATH<m>:STATe` on page 1456

Frequency axis (R&S RTO6-K37 only)

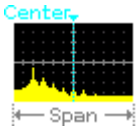
Defines the scaling method for the frequency (x-) axis of the spectrogram.

"Logarithmic" Logarithmic scaling

"Linear Unit" Linear scaling

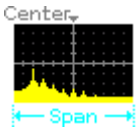
Remote command:
`CALCulate:MATH<m>:FFT:LOGScale` on page 1566

Center frequency
Defines the position of the displayed frequency range, which is (Center - Span/2) to (Center + Span/2). The width of the range is defined using the "Frequency span" setting.



Remote command:
`CALCulate:MATH<m>:FFT:CFRequency` on page 1567

Frequency span
The span is specified in Hertz and defines the width of the displayed frequency range, which is (Center - Span/2) to (Center + Span/2). The position of the span is defined using the "Center frequency" setting.



Remote command:
`CALCulate:MATH<m>:FFT:SPAN` on page 1568

Full span
Displays the full frequency span.

Remote command:
`CALCulate:MATH<m>:FFT:FULLspan` on page 1567

Start frequency
Defines the start frequency of the displayed frequency span.

Remote command:
`CALCulate:MATH<m>:FFT:STARt` on page 1566

Stop frequency
Defines the stop frequency of the displayed frequency span.

Remote command:
`CALCulate:MATH<m>:FFT:STOP` on page 1567

Span/RBW Coupling
Couples the frequency span to the "Resolution BW" setting.

Remote command:
`CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:AUTO` on page 1569

Span/RBW Ratio
Defines the coupling ratio for Span/RBW. This setting is only available if `CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:AUTO` is ON.

Remote command:
`CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:RATio` on page 1569

Resolution BW
Defines the resolution bandwidth. Note that the resolution bandwidth is correlated with the span, record length and acquisition time. If a constant record length is to be used, the RBW may be adapted if the required number of samples cannot be acquired. If span and RBW values are coupled, changing the span will also change the RBW.

For details see Chapter 9.1.1, "Fundamentals of FFT analysis", on page 370.

Remote command:
`CALCulate:MATH<m>:FFT:BANDwidth[:RESolution][:VALue]` on page 1569
`CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:ADJusted?` on page 1568

Window type
Windowing helps minimize the discontinuities at the end of the measured signal interval and thus reduces the effect of spectral leakage, increasing the frequency resolution.

Various different window functions are provided in the R&S RTO6 to suit different input signals. Each of the window functions has specific characteristics, including some advantages and some trade-offs. Consider these characteristics carefully, to find the optimum solution for the measurement task.

Window type	Frequency resolution	Magnitude resolution	Measurement recommendation
Rectangular	Best	Worst	Separation of two tones with almost equal amplitudes and a small frequency distance
Hamming Hann	Good	Poor	Frequency response measurements, sine waves, periodic signals and narrow-band noise
Blackman Harris (default)	Worst	Best	Mainly for signals with single frequencies to detect harmonics Accurate single-tone measurements
Gaussian	Good	Good	Weak signals and short duration
Flattop2	Poor	Best	Accurate single-tone measurements
Kaiser Bessel	Poor	Good	Separation of two tones with differing amplitudes and a small frequency distance

Remote command:
`CALCulate:MATH<m>:FFT:WINDow:TYPE` on page 1569

Use color table
If enabled, the spectrum waveform (and a spectrogram, if available) is displayed according to the assigned color table. For information on the available color tables, see Chapter 4.3.2, "Color tables", on page 76.

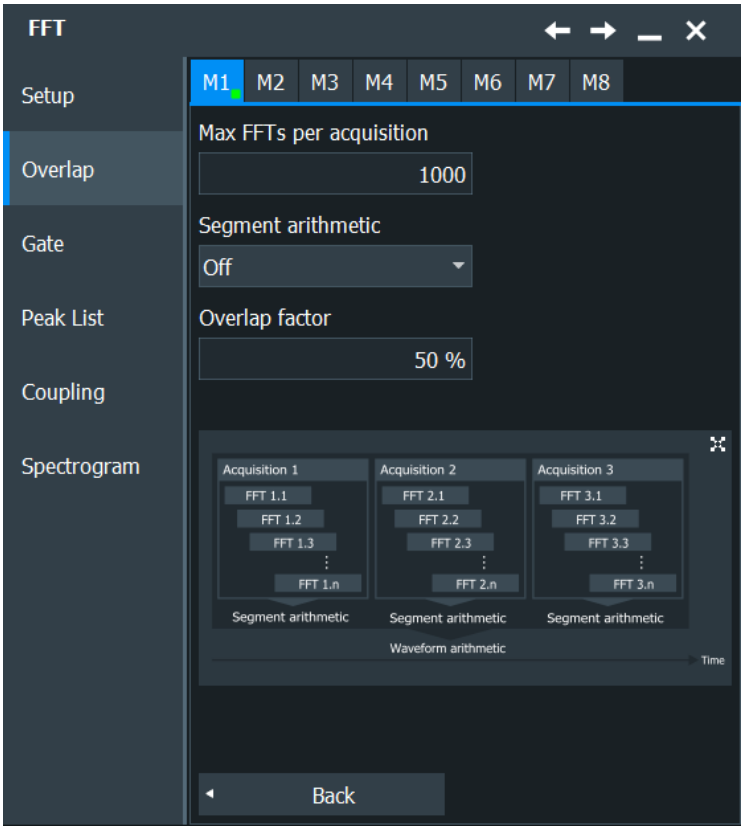
If this option is disabled, the preset color of the selected channel source is displayed, and the intensity of the specific signal color varies according to the cumulative occurrence of the values. For spectrum diagrams, this setting corresponds to the common waveform display. The spectrogram, on the other hand, is then displayed in gray tones, which is not useful.

Remote command:
`CALCulate:MATH<m>:FFT:USEColtab` on page 1580

9.1.3.2 FFT overlap

Access: [FFT] > "FFT setup" > "Overlap" tab

In this tab, you define the settings for the magnitude and phase of the frequencies.



Max FFTs per acquisition 382
Segment arithmetic 383
Overlap Factor 383

Max FFTs per acquisition
Restricts the maximum number of FFTs to be calculated for each data acquisition. Due to the other parameter settings, the required number of FFTs may become very high, thus slowing performance. By restricting the number of FFTs, you can avoid performance loss without changing the other parameters.

However, if the maximum number of FFTs is lower than the required number to cover the entire waveform, the waveform may only be analyzed partially. In this case, the "Frame coverage" indicates the percentage of the waveform that was analyzed, i.e. which part of the data was included in the FFT calculation.

Remote command:
`CALCulate:MATH<m>:FFT:FRAMe:MAXCount` on page 1571
`CALCulate:MATH<m>:FFT:FRAMe:COVerage?` on page 1571

Segment arithmetic
FFT analysis can only be performed on a maximum number of values at once. If more values must be calculated, the input signal is divided into segments, each of which is calculated separately. The segments need not be disjunct, that is: they may overlap, so that some values have several FFT results. In this case, the arithmetic mode defines how the final result is calculated from the individual results.

The following methods are available:

"Off" The data of only one segment is considered. In effect, no arithmetics are processed.

"Envelope" Detects the minimum and maximum values for FFT calculation over all segments. The resulting diagram shows two envelope waveforms: the minimums (floor) and maximums (roof). These envelopes indicate the range of all FFT values that occurred.

"Average" The average is calculated over all segments.

"RMS" The root mean square is calculated over all segments. The result is the average power spectrum. If you measure the channel power on this RMS spectrum, you get the same result as for the average channel power measurement on segments.

"MinHold" Determines the minimum result for each input value from the data of the current acquisition and the acquisitions before. Only available if option R&S RTO6-K37 is installed.

"MaxHold" Determines the maximum result for each input value from the data of the current acquisition and the acquisitions before. Only available if option R&S RTO6-K37 is installed.

Remote command:
`CALCulate:MATH<m>:FFT:FRAMe:ARIThmetics` on page 1571

Overlap Factor
Defines the minimum factor by which two neighboring segments overlap. If the required number of segments to cover the input values allows for more overlap, the factor is increased.

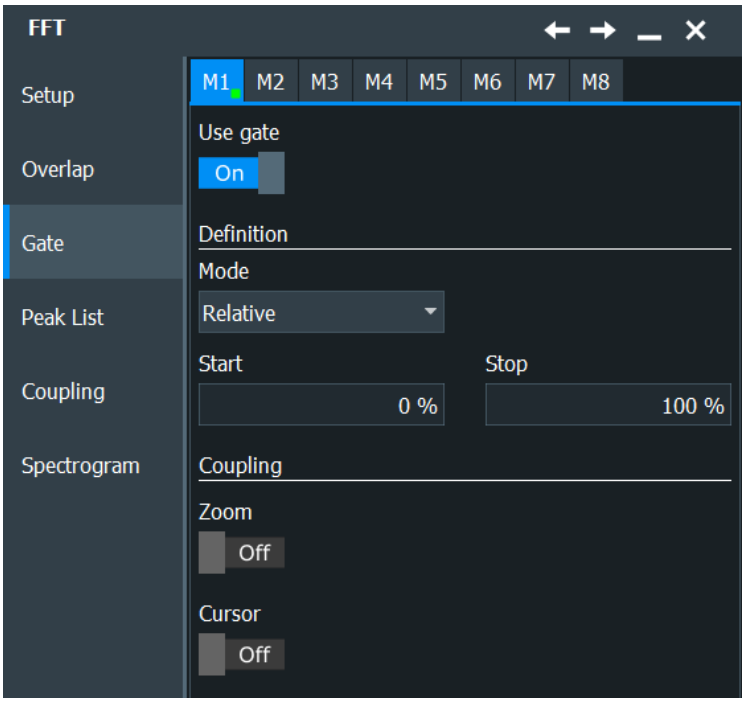
The higher the overlap factor, the more segments are used. This leads to more individual results and improves detection of transient signal effects. However, it also extends the duration of the calculation.

Remote command:
`CALCulate:MATH<m>:FFT:FRAMe:OFACtor` on page 1572

9.1.3.3 FFT gating

Access: [FFT] > "FFT setup" > "Gate" tab

FFT gating allows you to restrict FFT analysis to a certain time base of the input signal.



If no gate is used, you can define the record length as dependent on the RBW, or the RBW as dependent on the record length (which is defined by the acquisition time).

Use Gate....384
Gate Definition....384
└ Zoom....385
└ Gate description....385
└ (Relative) Start....385
└ (Relative) Stop....385
Record Length/RBW Coupling....386
Required acquisition time....386

Use Gate
Enables FFT gating and shows the gate.

If enabled, the "Gate Definition" settings are used.

If disabled, the relation between the record length and the RBW can be defined manually in the "Setup" tab instead.

When a gate is used, the RBW is adapted, if necessary. The smaller the gate, the higher the RBW.

For details, see Chapter 9.1.1, "Fundamentals of FFT analysis", on page 370.

Gate Definition
Defines the gate settings for FFT gating.

Zoom ← Gate Definition
Zoom coupling is available if a zoom is defined. As long as "Zoom coupling" is enabled, the gate area is defined identically to the zoom area - if you change the zoom, the gate changes as well.

If several zoom diagrams are defined, select the zoom diagram to be used for gating. The "Start" and "Stop" values of the gate are adjusted accordingly.

Zoom coupling can be set for measurement gates, FFT gates, and search gates. The zoom must be defined on the diagram that contains the signal source of the measurement, FFT, or search.

Remote command:
`MEASurement<m>:GATE:ZCOupling` on page 1548
`MEASurement<m>:GATE:ZDIagram` on page 1549
`CALCulate:MATH<m>:FFT:GATE:ZCOupling` on page 1574
`SEARch:GATE:ZCOupling` on page 1633
`SEARch:GATE:ZDIagram` on page 1634

Gate description ← Gate Definition
Defines whether the gate settings are configured using absolute or relative values.

"Absolute"	The gate is defined by absolute start and stop values.
"Relative"	The gate's start and stop values are defined by a percentage of the value range.

Remote command:
`CALCulate:MATH<m>:FFT:GATE:MODE` on page 1573
`MEASurement<m>:GATE:MODE` on page 1547
`SEARch:GATE:MODE` on page 1632

(Relative) Start ← Gate Definition
Defines the starting value for the gate.

Remote command:
`CALCulate:MATH<m>:FFT:GATE:ABSolute:STARt` on page 1573
`CALCulate:MATH<m>:FFT:GATE:RELative:STARt` on page 1574
`MEASurement<m>:GATE:ABSolute:STARt` on page 1547
`MEASurement<m>:GATE:RELative:STARt` on page 1547
`SEARch:GATE:ABSolute:STARt` on page 1632
`SEARch:GATE:RELative:STARt` on page 1633

(Relative) Stop ← Gate Definition
Defines the end value for the gate.

Remote command:
`CALCulate:MATH<m>:FFT:GATE:ABSolute:STOP` on page 1573
`CALCulate:MATH<m>:FFT:GATE:RELative:STOP` on page 1574
`MEASurement<m>:GATE:ABSolute:STOP` on page 1547
`MEASurement<m>:GATE:RELative:STOP` on page 1547
`SEARch:GATE:ABSolute:STOP` on page 1632
`SEARch:GATE:RELative:STOP` on page 1633

Record Length/RBW Coupling
The record length and resolution bandwidth are coupled during FFT analysis. If you change one value, the other must be adapted accordingly. You can keep either value constant, thus preventing automatic adaptation when the other parameter is changed. However, this may cause the FFT analysis to fail.

This setting is only available if gating is not enabled (otherwise the gate determines the RBW automatically).

For details, see Chapter 9.1.1, "Fundamentals of FFT analysis", on page 370.

"Record length controlled"	The record length remains constant. If not enough samples are available for the selected RBW, the RBW is decreased.
"RBW controlled"	The RBW is not adapted, i.e. remains as defined. The required acquisition time for this RBW is indicated. If necessary and possible, the record length is extended to acquire the required number of samples.

Remote command:
`CALCulate:MATH<m>:FFT:GATE:COUPling` on page 1572

Required acquisition time
The required acquisition time is calculated for the defined RBW value if "RBW constant" is selected, and is displayed for information only. If the required acquisition time is not available (e.g. because acquisition has already been stopped), an error message is displayed in the FFT setup tab indicating that not enough samples are available for the defined RBW.

Remote command:
`TIMebase:RACTime?` on page 1573

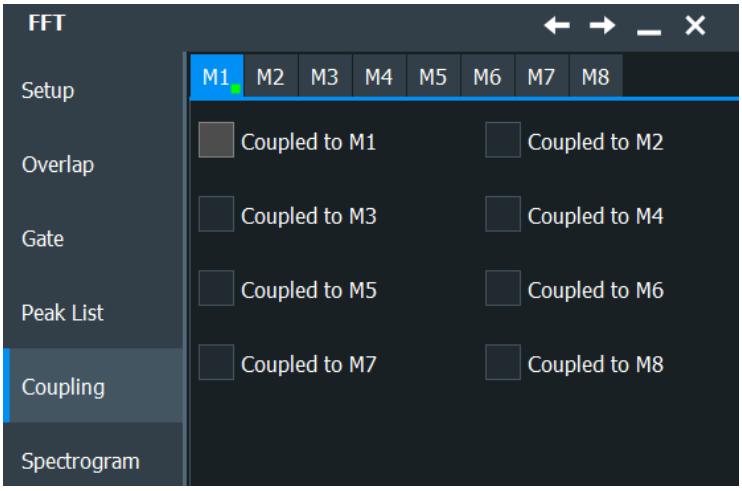
9.1.3.4 Peak list

For a description of the "Peak List" settings, see Chapter 8.2.7, "Spectrum measurements", on page 334.

9.1.3.5 FFT coupling

Access: [FFT] > "FFT setup" > "Coupling" tab

Up to four spectrum displays can be shown simultaneously, one for each math waveform. The settings for one or more spectrums can be coupled. Thus, if any FFT setting for any of the coupled spectrums are changed, they are changed for all coupled spectrums.



Coupled to M1...M8
Copies the current FFT settings of the selected math waveform (M1...M8) to the other selected math waveforms, and couples those waveforms. Thus, if any FFT setting for any of the coupled spectrums are changed, they are changed for all coupled spectrums.

Two different sets of spectrums can be coupled at the same time, for instance "M1" can be coupled to "M2", while "M3" is coupled to "M4".

Note that the formula of the coupled math waveforms is not changed. If necessary, you must select an FFT function for the math waveform manually before the FFT settings of the coupled waveform are applied. See Chapter 7.3.1, "Displaying math waveforms", on page 257.

Remote command:
`CALCulate:MATH<m>:FFT:COUPled:WITH<1..8>` on page 1578

9.1.4 FFT scale settings

In this section, you can find functions to set the vertical parameters of the FFT math waveform.

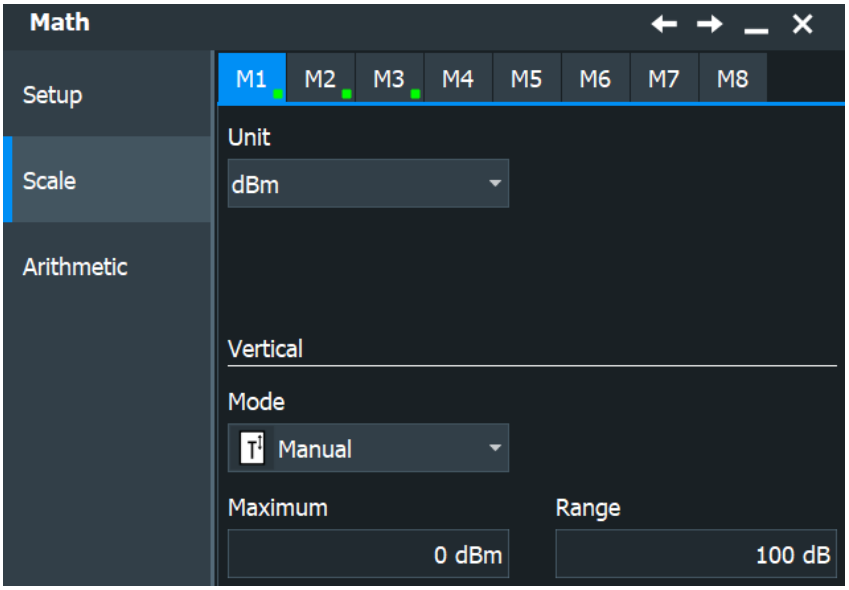


Figure 9-1: Scale settings of magnitude spectrum

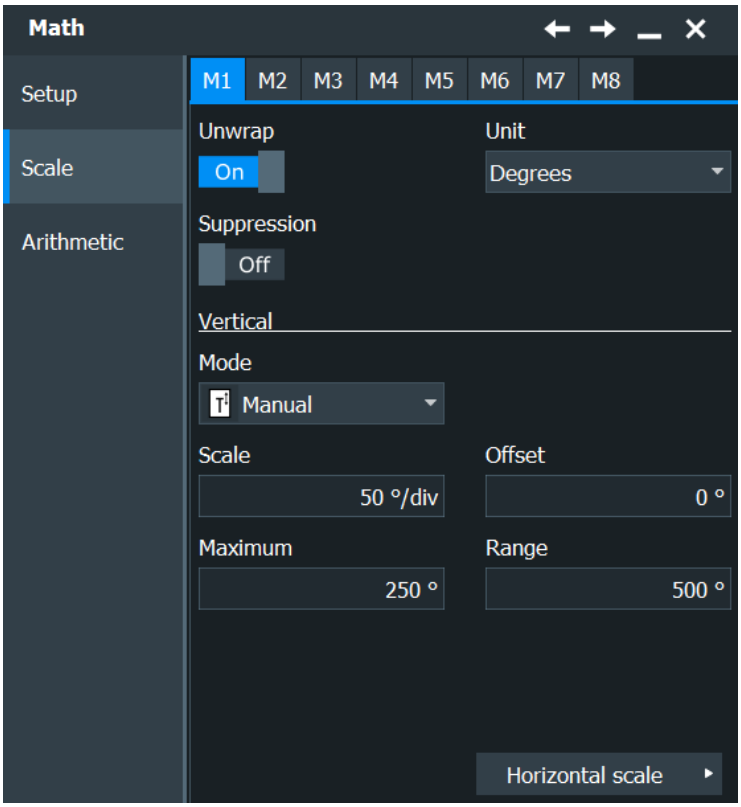


Figure 9-2: Scale settings of phase spectrum

See also:

- Vertical scaling mode (Manual/Auto)
- Vertical Offset

- Vertical Scale

Unit
Defines the scaling unit for magnitude display. Available units depend on the unit of the selected source waveform.

"Linear" displays the RMS value of the voltage.

Remote command:
`CALCulate:MATH<m>:FFT:MAGNitude:SCALe` on page 1575

Reference level
Defines the reference level if "dB" is the selected unit.

Remote command:
`CALCulate:MATH<m>:FFT:MAGNitude:LEVel` on page 1575

Vertical maximum
Defines the maximum value on y-axis for spectrum displays. Only available for "Manual" scale mode.

Remote command:
`CALCulate:MATH<m>:VERTical:MAXimum` on page 1459

Vertical range
Defines the range of FFT values to be displayed.

Remote command:
`CALCulate:MATH<m>:VERTical:RANGe` on page 1458

Unwrap
If enabled, phase shifts due to a limitation of the value range are eliminated.

Remote command:
`CALCulate:MATH<m>:FFT:PHASe:UNWRap` on page 1577

Unit
Defines the scaling unit for phase display.

- Radians
- Degrees

Remote command:
`CALCulate:MATH<m>:FFT:PHASe:SCALe` on page 1577

Suppression
Enables noise suppression. Phase calculation is restricted to frequencies with a minimum magnitude, the threshold value.

Remote command:
`CALCulate:MATH<m>:FFT:PHASe:SUPPression` on page 1577

Threshold
Defines the minimum frequency magnitude for which phases are calculated. This setting is only available if "Suppression" is enabled.

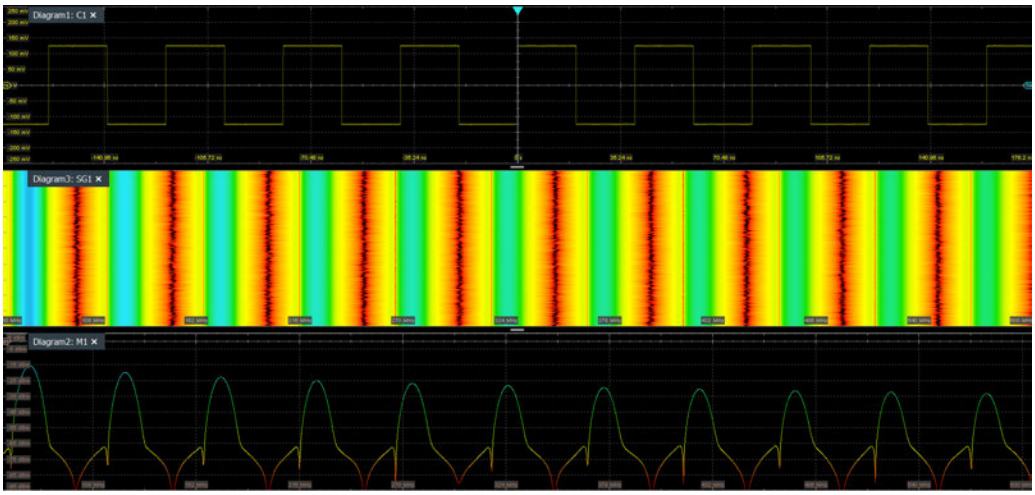
Remote command:
`CALCulate:MATH<m>:FFT:PHASe:THReshold` on page 1577

9.2 Spectrum analysis (option R&S RTO6-K37)

This chapter describes the Spectrum analysis option R&S RTO6-K37.

9.2.1 Spectrogram display

The Spectrum analysis option provides a new diagram for spectrum waveforms: a spectrogram. When you enable a spectrogram, three windows are displayed: the power vs. time diagram at the top, the spectrogram in the middle (labeled "SG") and the power vs. frequency (=spectrum) diagram at the bottom.



A spectrogram shows how the spectral density of a signal varies over time. The x-axis shows the frequency, the y-axis shows the time. A third dimension, the power level, is indicated by different colors. Thus you can see how the strength of the signal varies over time for different frequencies.

The spectrogram is updated with each data acquisition, from top to bottom, so that the most recent trace is at the bottom. Up to two time lines can be shown at a specified position so that you can analyze the spectrum at a specific point in time.

The spectrum diagram indicates the power vs. frequency values for a single data acquisition. If a time line is enabled, the spectrum shows the results at the selected time. Otherwise, the spectrum shows the results of the most recent data acquisition.

9.2.2 Spectrum analysis functions

In addition to spectrograms, the Spectrum analysis option also provides some new automatic measurements based on spectrum waveforms.

- A peak list measurement detects all peaks above a user-definable threshold and optionally indicates the peaks in the spectrum diagram.

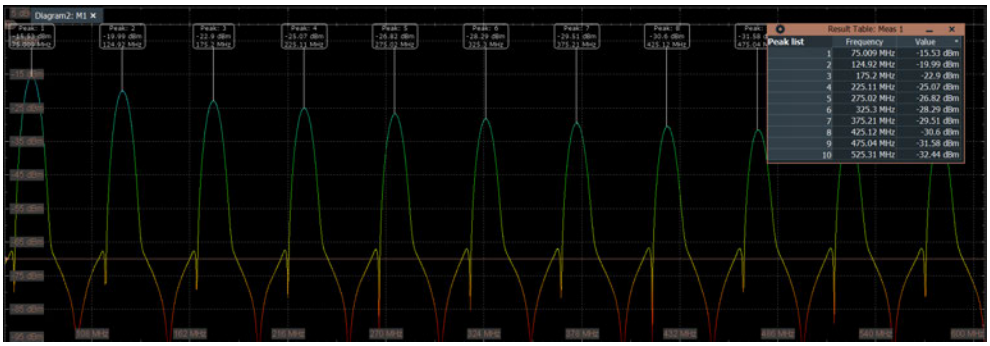


Figure 9-3: Peak list with labels for spectrum waveform

  For a description of the measurement settings, see Chapter 8.2.7.2, "Settings for spectrum measurements", on page 336.
- The THD measurements are an extension to the basic THD measurement. See Chapter 8.2.7, "Spectrum measurements", on page 334 for details.
- Cursor measurements on spectrum waveforms provide easy center definition and peak search functions, see Chapter 8.1.1.2, "Cursor measurements on spectrum waveforms", on page 285.

9.2.3 Configuring spectrograms

Spectrograms are only available if the Spectrum analysis option R&S RTO6-K37 is activated.

1. Tap the "Spectrogram" icon on the toolbar.

   The "Spectrogram" overlay menu opens.

2. Select the source of the spectrogram.

   A spectrogram diagram is displayed. A new signal icon for the spectrogram is displayed on the signal bar ("SGx").
   If the selected source is a channel waveform, an FFT is started, on which the spectrogram is created.
   Additional settings for time lines become available in the dialog box.

3. Optionally, to display a time line and thus mark a specific waveform in the spectrogram, select "Enable" for one of the two time lines.

A small arrow icon labeled "T1" / "T2" indicates the position of the time line in the spectrogram.
The spectrum diagram displays the results for the selected waveforms. A new signal icon is displayed on the signal bar for each time line ("SGxTL1|2").

4. To view the spectrum for each time line in a separate diagram, drag the signal icon for one time line to the diagram area and drop it.

   A new window is displayed for the selected time line, and the original diagram displays the other time line.

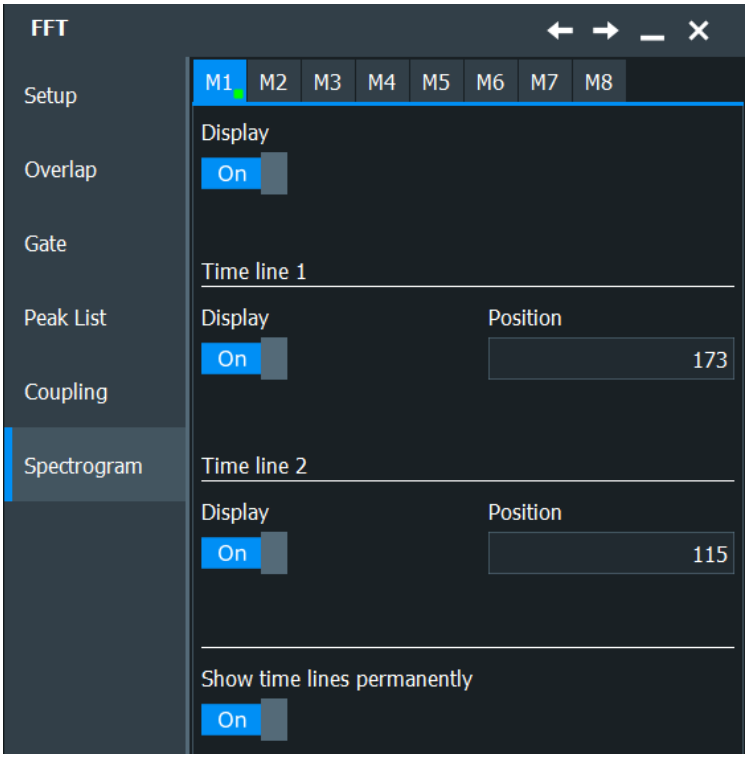
5. To view a different waveform from the spectrogram, move the time lines in the spectrogram.

9.2.4 Spectrogram configuration settings

Access: [FFT] > "FFT setup" > "Spectogram" tab

Spectrograms are only available if the option R&S RTO6-K37 is activated. Furthermore, a math (FFT) waveform must be configured and enabled.

See Chapter 9.1.2, "Configuring spectrum waveforms", on page 374.



Display 393
Time line 1/2 393
Show lines permanently 393

Display
Enables the spectrogram display.

If enabled, a new signal icon for the spectrogram is displayed on the signal bar ("SGx").

Remote command:
`CALCulate:MATH<m>:FFT:SPECtrogram:STATe` on page 1581

Time line 1/2
A time line marks a single spectrum in the spectrogram, that is: the power vs frequency results for the data acquired at a specific time. After enabling a time line, the results for that time are displayed in the spectrum diagram. A small arrow icon labeled "T1" / "T2" indicates the position of the time line in the spectrogram. A new signal icon is displayed on the signal bar for each time line ("SGxTL1|2").

You can enable and display two time lines at the same time. This allows you to compare the results at different times.

The position of the time line is defined by the index of the data acquisition in the history. How many acquisitions are available depends on the history settings.

See "Player" on page 276.

Remote command:
`CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:STATe` on page 1581
`CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:POSition` on page 1581

Show lines permanently
Displays the spectrogram time lines in the diagrams until you disable this option.

If disabled, only the small arrow icons are permanently visible. The line is only displayed temporarily when you touch the arrow.

10 Mask testing

10.1 About mask testing

Masks are used to determine whether the signal remains within specified limits, e.g. to uncover signal anomalies or test compliance and stability of digital signals. The limits are specified as "mask", which is laid over the input signal in the display. Thus you can easily detect where the signal violates the mask.

Mask testing with R&S RTO6 has only a minor impact on the acquisition rate, thus mask violations are detected fast and reliably.

With R&S RTO6, you can define own masks easily. Specific actions can be executed when mask violations occur. For error analysis, you can stop the acquisition on a failed test and use the history view to look at the previous waveforms.

Mask test

A mask test consists of:

- Mask definition
- Waveform to be tested
- Fail criteria for test
- Actions to be taken on violation or successful completion

Mask Definition

A mask can be created in several ways:

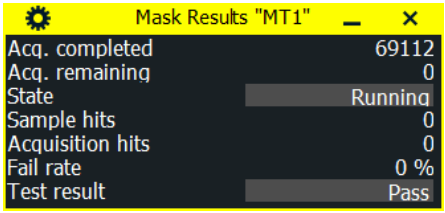
- The individual mask points are defined, either on the touch screen or as numerical values. This mask type is called *user mask*.
  For details, see Chapter 10.2.2.1, "Mask definition: user mask", on page 398
- The mask is derived from an existing waveform. This mask type is called *waveform mask*.
  For details, see Chapter 10.2.2.2, "Mask definition: waveform mask", on page 401.

Fail Criteria for Testing

The fail criteria for a mask test is set by two parameters: "Fail condition" and "Violation tolerance". Fail condition defines if sample hits or the number of acquisitions with sample hits are considered. Violation tolerance sets the number of tolerable sample hits or acquisition hits. A test has failed if the number of sample hits or acquisition hits exceeds the limit of violation tolerance hits.

10.1.1 Results of a mask test

The result box of a mask test shows the following test results:



Acq. completed
Number of tested acquisitions.

Remote command:
`MTESt:RESult:COUNt:WAVeforms?` on page 1597

Acq. remaining
Remaining acquisitions until "Average count / Nx Single count" is reached.

The value is useful if you test a specified number of acquisitions with action "Stop acquisition" on violation. Also if the acquisition has been stopped manually before the required number of acquisitions has been acquired.

Remote command:
`MTESt:RESult:COUNt:REMaining?` on page 1598

State
Shows if the test has been completed. The state is set to "Finished" when "Nx Single count" acquisitions are tested and the number of "Acq. remaining" is 0. as long as the number of tested acquisitions is less the "Nx Single count" number, the state is "Running".

If you run the acquisition with [Run Stop], or the number of played history acquisitions exceeds "Nx Single count", the mask testing is performed according to fail criteria settings independently of the test state. The testing is not stopped when the state is set to "Finished".

Remote command:
`MTESt:RESult:STATe?` on page 1597

Sample hits
Number of samples that hit the mask.

Remote command:
`MTESt:RESult:COUNt:FAILures?` on page 1598

Acquisition hits
Number of acquisitions that contained at least one sample hit.

Remote command:
`MTESt:RESult:COUNt:FWAVeforms?` on page 1598

Fail rate
Ratio of acquisition hits to the number of tested acquisitions.

Remote command:
`MTESt:RESult:FRATe?` on page 1599

Test result
A test has failed if the number of sample hits or acquisition hits exceeds the limit of "Violation tolerance" hits.

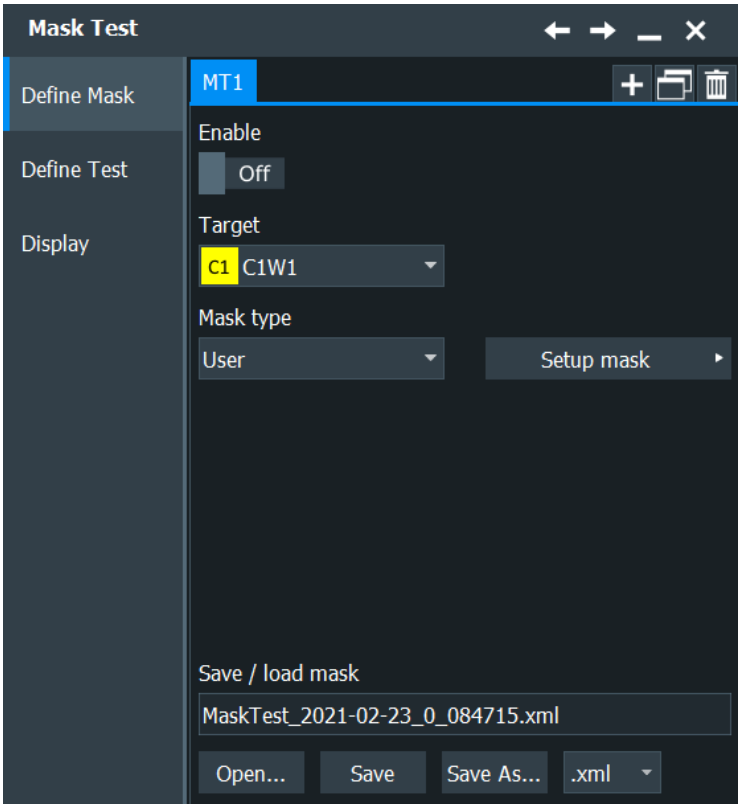
Remote command:
`MTESt:RESult[:RESult]?` on page 1597

10.2 Mask test settings

10.2.1 Define mask

Access: [App Cockpit] > "Analysis" tab > "Mask" > "Define Mask"

The "Define Mask" tab provides all settings for the mask test itself: the waveform to be tested,and saving/loading the mask definition.



The content of the "Test Definition" tab depends on the selected definition type. If "Waveform" is selected, the main mask settings can be set directly on the "Test Definition" tab. For a description of these settings, see Chapter 10.2.2.2, "Mask definition: waveform mask", on page 401.

Make sure that the correct "Mask Test" tab is selected on the left side before you enter the settings.

Remote commands:

`MTESt:ADD` on page 1582

`MTESt:REMove` on page 1583

Enable
Activates and deactivates the mask test. If the acquisition is running, the test starts immediately. Otherwise, the test starts when acquisition is started.

The testing is stopped when acquisition is stopped, or if a stop action is configured with "Stop acq." on page 406.

Closing the result box also disables the mask test.

Remote command:
`MTESt[:STATe]` on page 1583

Target
Selects the waveform to be tested against the mask. All channel waveforms can be tested.

Remote command:
`MTESt:SOURce` on page 1584

Mask Type
Sets the method of mask definition.

"User"	The mask is created manually by tapping the mask points on the touch screen and/or by entering the numerical x- and y-values of the mask points.
"Waveform"	The mask is created from an existing waveform. The waveform builds the upper and lower limit line of the mask, and the limits are moved and stretched. The result is a tolerance tube around the waveform that is used as mask.
"Eye"	Requires jitter option R&S RTO6-K12. The mask is created by selecting the shape and setting its dimensions according to the test standard.

Remote command:
`MTESt:CTYPe` on page 1585

Setup mask
Opens a dialog for defining the mask. See Chapter 10.2.2, "Mask definition", on page 398.

Save / load mask
Provides all functions to store and recall a mask test. The mask definition, defined actions and fail conditions are stored in an R&S RTO6-specific .xml file.

"Open" Opens a dialog box where you can select the file name.
See also: Chapter 12.5, "File selection dialog", on page 482.

"Save, Save As" Stores the specified file.

Remote command:
`MTESt:FILE:NAME` on page 1585
`MTESt:FILE:SAVE` on page 1586
`MTESt:FILE:OPEN` on page 1586
`MTESt:FILE:DELete` on page 1586

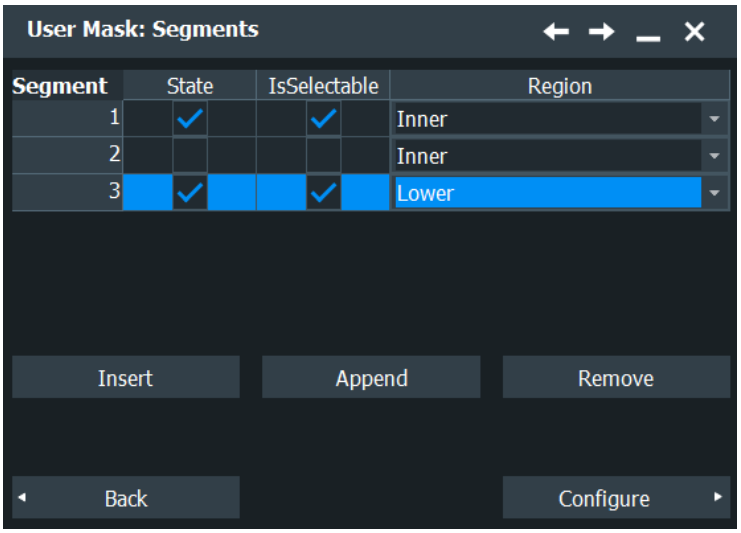
10.2.2 Mask definition

With mask definition, you define the shape of the mask - the form and position of its limit lines. The content of the "Mask Definition" tab depends on the selected Mask Type: "User" or "Waveform".

Below, you find the specific settings:

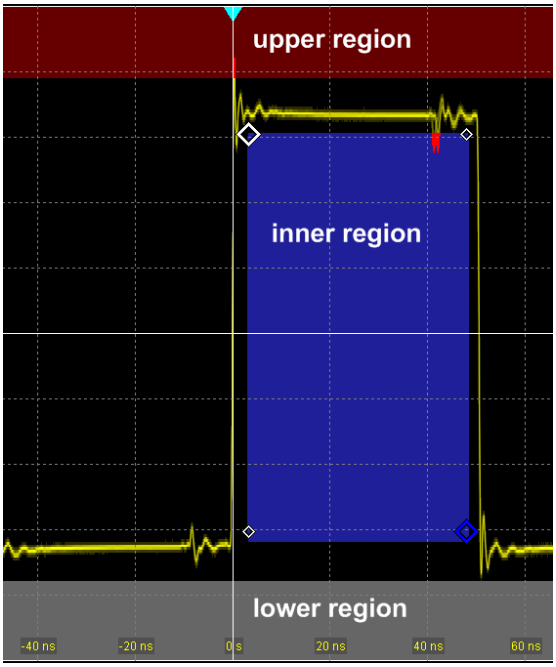
10.2.2.1 Mask definition: user mask

Access: [App Cockpit] > "Analysis" tab > "Define Mask" > "Mask Type" = "User" > "Setup mask"



A user mask is defined by entering the time and voltage values for all corner points of the mask segments. A user mask has at least one segment. Complex masks can have up to 16 segments.

An inner segment is an area defined by three or more points. Upper and lower segments limit the signal on top and bottom of the screen. They are defined by a line, the region above or below the line is set automatically as mask segment.



Alternatively, you can set the corner points on the touch screen and adjust the values in the "Mask Definition" tab.

To save the mask, select the "Define Mask" tab and save the mask test.

Make sure that the correct "Mask Test" tab is selected on the left side before you enter the settings.

Mask segments

Defines the number and state of mask segments for the selected mask test. Here you can:

- Insert a new segment before the selected segment.
- Append a new segment at the end of the list.
- Remove the selected mask segment from the list.
- Select the region that builds the mask.
  - Inner region: the segment points form a closed geometrical shape, which is the mask segment.
  - Upper region: the segment points are connected to a line, the display area above this line is the mask segment.
  - Lower region: the segment points are connected to a line, the display area below this line is the mask segment.
- Enable and disable the mask segments individually. Disabled segments are not considered by running tests.

Remote command:

`MTESt:SEGMent:STATe` on page 1587
`MTESt:SEGMent:ADD` on page 1587
`MTESt:SEGMent:REMove` on page 1588
`MTESt:SEGMent:INSert` on page 1587

MTESt:SEGMent:REGion on page 1588
MTESt:SEGMent:COUNt? on page 1587

Configure
Opens a dialog to define the selected segment.

Definition of segment
The number of the selected segment is shown above the table. In the definition table, the individual points of the selected mask segment are listed with exact horizontal and vertical numerical coordinates. Here you can:

- Insert a new point before the selected point.
- Append a new point at the end of the list.
- Remove the selected point from the list.
- Change the x- and y-values of each point. To scale or move the complete segment, use offset and factor values, see Rescale.

Remote command:
MTESt:SEGMent:POINt:ADD on page 1589
MTESt:SEGMent:POINt:REMove on page 1589
MTESt:SEGMent:POINt:INSert on page 1589
MTESt:SEGMent:POINt:X on page 1590
MTESt:SEGMent:POINt:Y on page 1590
MTESt:SEGMent:POINt:COUNt? on page 1589

Rescale
You can rescale and move mask segments by numerical input of factors and offsets.

The values change the selected mask segment and take effect on "Apply".

Offset X ← Rescale
Moves the mask segment horizontally. The specified offset is added to the x-values of all points of the selected mask segment.

To take effect, tap "Apply".

Remote command:
`MTESt:SEGMent:RESCale:XOFFset` on page 1591

Factor X ← Rescale
Stretches or compresses the selected mask segment in horizontal direction. The x-values of all points of the selected mask segment are multiplied with this factor. Factors >1 stretch the mask segment, while factors between 0 and 1 compress it. Negative values are possible and change the algebraic sign.

To take effect, tap "Apply".

Remote command:
`MTESt:SEGMent:RESCale:XFACtor` on page 1591

Offset Y ← Rescale
Moves the mask segment vertically. The specified offset is added to the y-values of all points of the selected mask segment.

To take effect, tap "Apply".

Remote command:
`MTESt:SEGMent:RESCale:YOFFset` on page 1592

Factor Y ← Rescale
Stretches or compresses the selected mask segment in vertical direction. The y-values of all points of the selected mask segment are multiplied with this factor. Factors >1 stretch the mask segment, while factors between 0 and 1 compress it. Negative values are possible and change the algebraic sign.

To take effect, tap "Apply".

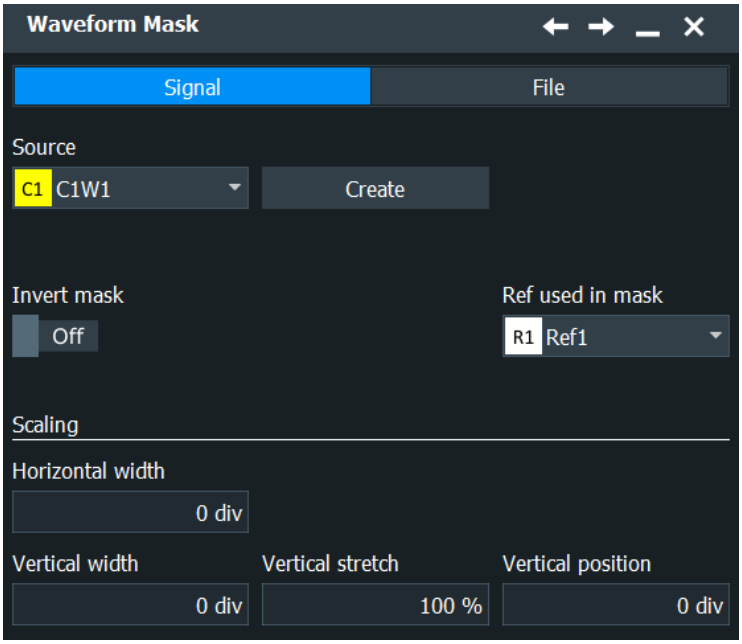
Remote command:
`MTESt:SEGMent:RESCale:YFACtor` on page 1591

Apply ← Rescale
Multiplies and adds the given x- and y-factors and offsets to the coordinates of all points of the selected mask segment.

Remote command:
`MTESt:SEGMent:RESCale:RECalculate` on page 1590

10.2.2.2 Mask definition: waveform mask

Access: [App Cockpit] > "Analysis" tab > "Define Mask" > "Mask Type" = "Waveform" > "Setup mask"



A waveform mask is created from an existing waveform. The waveform builds the upper and lower limit line of the mask, and the limits are moved and stretched. The result is a tolerance tube around the waveform that is used as mask.

During mask testing using a waveform mask, the record length is limited to 1 MSample.

The source for a waveform mask is a reference waveform. The reference waveform can be defined before mask definition, or loaded from a file, or it is created from the waveform to be tested.

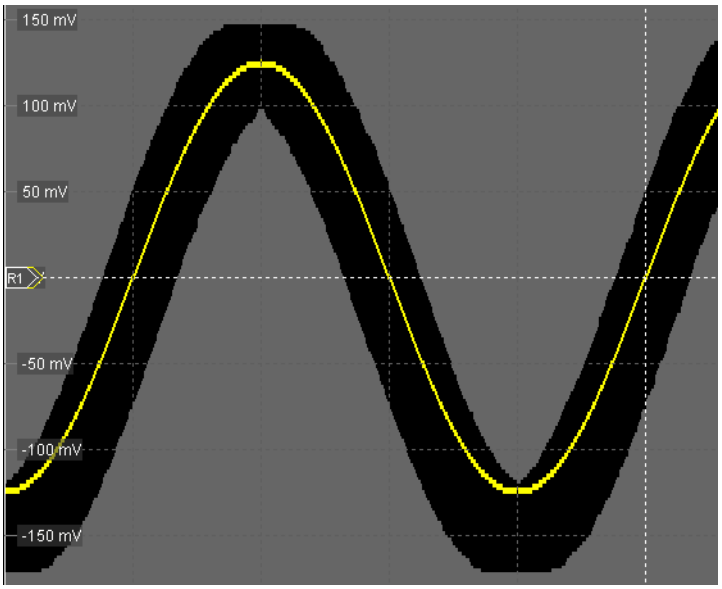


Figure 10-1: Waveform mask

Signal
The reference waveform is created from the selected "Source".

Sets the reference waveform from which the mask is created.

Create ← Signal
Creates the upper and lower mask limit from the "Source" and "Ref used in mark". If the reference waveform was not defined before, it is created automatically from the mask test "Source" waveform.

File
Loads the waveform from the selected file to the "Reference" and creates the mask immediately.

Invert mask
If enabled, the area inside the mask is highlighted, which the signal must not exit. If disabled, the area outside the mask limits is highlighted.

Horizontal width
Sets the width of the mask in horizontal direction. The specified number of divisions is added to the positive x-values and subtracted from the negative x-values of the mask limits in relation to the source waveform of the mask. The overall mask width is twice the specified horizontal width.

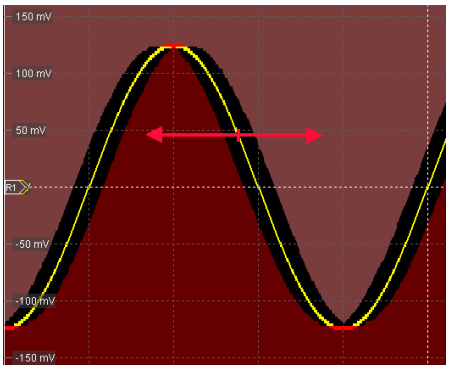


Figure 10-2: Waveform mask with horizontal width = 0.2 div

Remote command:
`MTESt:WFMRescale:XWIDth` on page 1593

Vertical width
Sets the width of the waveform mask in vertical direction. The specified number of divisions is added to the y-values of the upper mask limit and subtracted from the y-values of the lower mask limit. Thus, the upper half of the mask is pulled upwards, the lower half is pulled down, and the overall height of the mask is twice the vertical width.

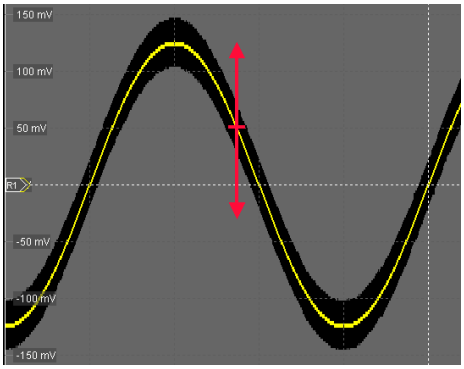


Figure 10-3: Waveform mask with vertical width = 0.5 div

Remote command:
`MTESt:WFMRescale:YWIDth` on page 1593

Vertical stretch
Sets the vertical scaling to stretch the mask in y-direction. The scaling axis is the horizontal line through the lowest value of the lower mask limit. Values > 100% stretch the mask, and values < 100% compress it.

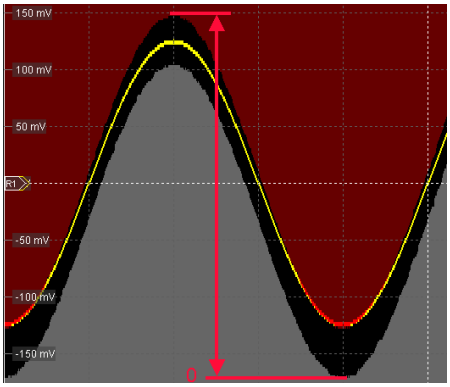


Figure 10-4: Waveform mask with vertical width = 0.5 div, vertical position = -0.5 div, vertical stretch = 110%

Remote command:
`MTESt:WFMRescale:YSTRetch` on page 1594

Vertical position
Moves the mask vertically within the display.

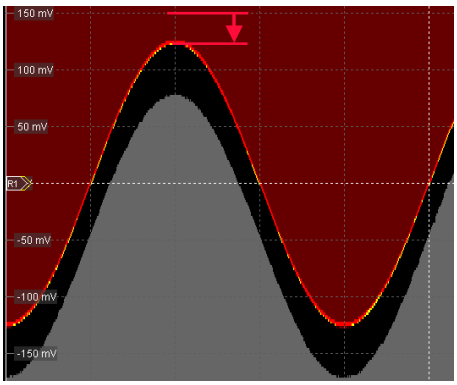
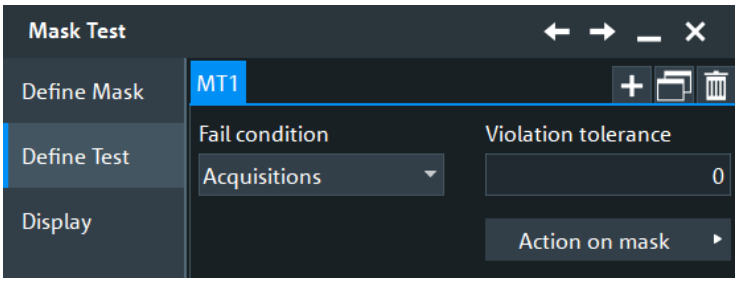


Figure 10-5: Waveform mask with vertical width = 0.5 div and vertical position = -0.5 div

Remote command:
`MTESt:WFMRescale:YPOSition` on page 1594

10.2.3 Define Test

Access: [App Cockpit] > "Analysis" tab > "Mask" > "Define Test"



There are two usual test practices:

- Testing a defined number of waveforms against the mask and initiate an action when the acquisition cycle has been completed without failure:
  - Set the number of acquisitions to be tested: "Average count (N-single count)"
  - Start [Single]
- Testing a continuous acquisition or a defined number of waveforms against the mask and initiate an action when the fail criteria is fulfilled.

Make sure that the correct mask test tab is selected before you enter the settings.

Fail condition, Violation tolerance
The fail criteria for a mask test is set by two parameters: "Fail condition" and "Violation tolerance".

"Fail condition" defines the kind of hits to be considered for test evaluation:

- "Samples": Considers the number of samples that hit the mask.

- "Acquisitions": Considers the number of acquisitions that contain at least one sample hit. How many samples hit the mask in that acquisition is not relevant.

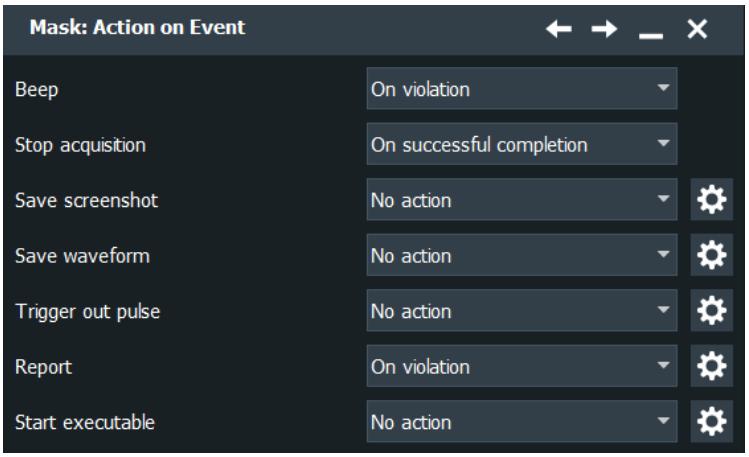
"Violation tolerance" sets the number of tolerable sample hits or acquisition hits.

A test has failed if the number of sample hits or acquisition hits exceeds the limit of violation tolerance hits.

Remote command:
`MTESt:CONDition` on page 1584
`MTESt:TOLerance` on page 1584

Action on mask
The settings in this dialog define what happens when the mask test has failed or when it has passed successfully.

Most actions can be initiated either on failure or on success:
- On violation
  The action is initiated when the fail criteria is fulfilled.
- On successful completion
  The action is initiated when the [Single] acquisition has finished and the fail criteria is not fulfilled - the fail condition and violation tolerance limit have not been reached.



Beep ← Action on mask
Generates a beep sound.

Remote command:
`MTESt:ONViolation:BEEP` on page 1594

Stop acq. ← Action on mask
Stops the waveform acquisition on mask violation.

Remote command:
`MTESt:ONViolation:STOP` on page 1595

Save screenshot ← Action on mask
Saves a screenshot including the mask test results according to settings in "Menu" > "Save/Recall" > "Save" tab > "Screenshot".

Remote command:
`MTESt:ONViolation:PRINt` on page 1595

Save waveform ← Action on mask
Saves the waveform data to a file according to settings in "Menu" > "Save/Recall" > "Save" tab > "Waveforms".

Remote command:
`MTESt:ONViolation:SAVewaveform` on page 1595

Trigger out pulse ← Action on mask
Sends a pulse to the [Trigger Out] connector on the rear panel.

You can adjust the pulse by tapping the settings icon, see "Trigger out signal setup" on page 225.

Remote command:
`MTESt:ONViolation:TRIGgerout` on page 1596

Report ← Action on mask
Creates and saves a report using the settings.

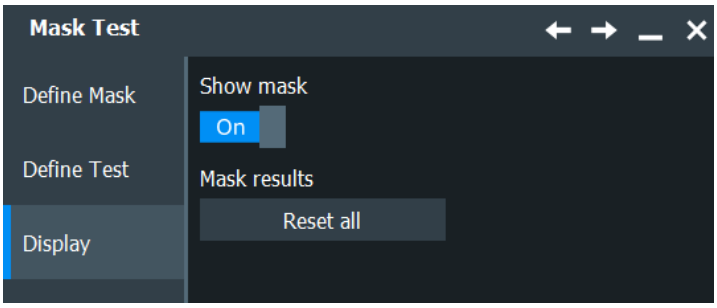
Remote command:
`MTESt:ONViolation:REPort` on page 1596

Start Executable ← Action on mask
Starts an external application. Tap "Config Executable" to set the application path and parameters.

Remote command:
`MTESt:ONViolation:RUNexec` on page 1596

10.2.4 Mask display

Access: [App Cockpit] > "Analysis" tab > "Mask" > "Display".



In this dialog, you can reset the mask results and display the mask.

Show mask
Switches the display of all mask segments on or off.

Remote command:
`MTESt:SHOW` on page 1583

Mask results > Reset all
Clears all totals and results in all "Mask Test" result boxes.

Remote command:
`MTESt:RST` on page 1583

10.2.5 Mask appearance

You can define additional mask appearance settings in the "Settings" > "Appearance" > "Mask" dialog.

See Chapter 4.3.10, "Mask appearance settings", on page 87

10.3 Mask testing on eye diagrams

To perform mask testing on eye diagrams, the option R&S RTO6-K12 adds a special mask definition type to the common mask definitions: the definition type "Eye".

Using the "Eye" mask type, you can easily define eye masks of various shapes. You can also use "User" masks to define free mask shapes.

10.3.1 Test definition for eye mask tests

Access: [App Cockpit] > "Analysis" tab > "Define Mask" tab

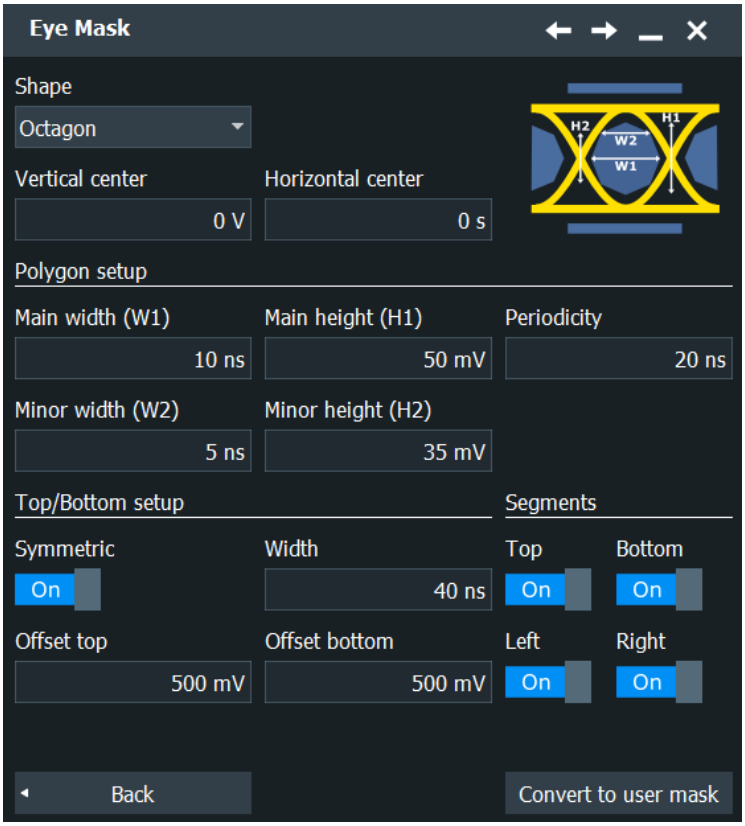
1. Select the "Source": the channel waveform that is analyzed for jitter.
2. Select the "Mask Type" = "Eye".
3. In the "Define Test" tab, adjust the "Fail condition" and "Violation tolerance".
4. Define the eye mask. See Chapter 10.3.2, "Eye mask definition", on page 408.

The general mask test settings are described in Chapter 10.2, "Mask test settings", on page 396.

10.3.2 Eye mask definition

Access: [App Cockpit] > "Analysis" tab > "Define Mask" > "Mask Type" = "Eye" > "Setup mask"

The "Eye Mask " dialog provides all settings to define masks of different shapes. It is available if option R&S RTO6-K12 is activated on the instrument.



Shape
Defines the outline of the eye mask: square, diamond, hexagon or octagon.

Remote command:
`MTESt:EYEMask:TYPE` on page 1600

Horizontal center/Vertical center
Set the horizontal (time) and vertical (voltage) values of the eye shape enter and thus define the position of the eye shape on the display.

Remote command:
`MTESt:EYEMask:HPOSition` on page 1603
`MTESt:EYEMask:VPOSition` on page 1603

Main width (W1), Minor width (W2)
Main width defines the width of all eye mask shapes. Minor width defines the secondary width for hexagon and octagon mask shapes.

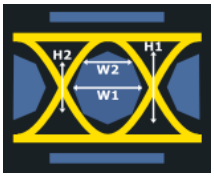


Figure 10-6: Main and minor widths and heights of an octagon eye mask

Remote command:
`MTESt:EYEMask:WIDTh<m>[:VALue]` on page 1600

Main height (H1), Minor height (H2)
Main height defines the height of all eye mask shapes. Minor height defines the secondary height for octagon mask shapes.

If an octagon shape is selected, and both values are the same, the eye masks looks like a hexagon rather than an octagon.

Remote command:
`MTESt:EYEMask:HEIGht<m>[:VALue]` on page 1600

Periodicity
Sets the time distance between the shape centers.

Remote command:
`MTESt:EYEMask:HPERiod` on page 1601

Top/Bottom setup
Defines the settings for the top and bottom segments of the mask. You have to enable "Top" / "Bottom" for this settings to be relevant.

Symmetric ← Top/Bottom setup
Sets bottom and top offsets to the same value so that the outer regions are symmetric to the eye shape.

Remote command:
`MTESt:EYEMask:TBSYmmetric` on page 1602

Width ← Top/Bottom setup
Sets the time width of the outer regions, symmetric to the eye shape center.

Remote command:
`MTESt:EYEMask:TBWidth` on page 1602

Offset top, Offset bottom ← Top/Bottom setup
Sets the voltage distance from the eye shape center that limits the upper and lower regions.

Remote command:
`MTESt:EYEMask:BOFFset` on page 1602
`MTESt:EYEMask:TOFFset` on page 1602

Segments
Enable the different regions of the mask: "Top", "Bottom", "Left", "Right".

The icons copy the eye shape to the left and to the right.

Remote command:
`MTESt:EYEMask:MSKBottom` on page 1601
`MTESt:EYEMask:MSKTop` on page 1601
`MTESt:EYEMask:MSKLeft` on page 1601
`MTESt:EYEMask:MSKRight` on page 1601

Convert to user mask
Converts the test and mask definitions of the current mask test to a new mask test of type "User".

The new test is opened in a separate mask test tab, in the "Define Mask" tab.

This is useful, for example, if want to run variants of a mask test in parallel.

Remote command:
`MTESt:CEMask` on page 1599

10.4 Working with masks

This chapter explains step-by-step how mask tests are setup and preformed. For the explanation of the individual settings, see Chapter 10.2, "Mask test settings", on page 396.

- Setting up user masks....411
- Setting up a mask test....416
- Configuring the mask and hit display....416
- Running a mask test....417
- Saving and loading masks....417
- Mask testing on history acquisitions....418

10.4.1 Setting up user masks

10.4.1.1 Creating user masks

There are two ways to create a new mask:

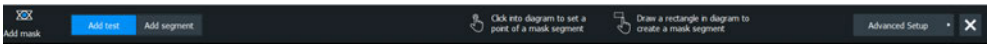
- Graphical way by tapping the mask points on the touchscreen,
- Numerical entry of the x- and y-values of the mask points.

You can combine both methods. For example, at first you enter the mask quickly on the touchscreen, and then modify the point coordinates with precise values.

To create a mask graphically on the touch screen

1. Tap the "Masks" icon on the toolbar.

   The "Masks" overlay menu opens.

2. Tap the corner points of the mask segment on the touch screen.

   Tip: To create an exact rectangle, draw the diagonal of the rectangle on the screen.

3. Tap "Finish segment".

   Now you can enter another segment to the current mask test or add a new mask test.

4. To finish the mask definition, close the overlay menu.
   If the acquisition is running, the mask test starts automatically and immediately.

To create a mask numerically in the dialog box

The settings mentioned here are described in detail in Chapter 10.2.1, "Define mask", on page 396.

1. Open [App Cockpit] > "Analysis" tab > "Mask" > "Define Mask" tab.
2. Create a mask test:
   a) Tap the "+"-icon in the upper right corner.
   b) Enter a name for the new mask test.

   A new, empty tab for the mask test appears.
3. Set "Mask type"= "User".
4. Tap "Setup mask".
5. In the "User Mask: Segments" area, tap "Insert" to create a new mask segment.
6. Set the corner points of the mask segment:
   a) Select the "Segment".
   b) Tap "Configure". The "User Mask: Points" dialog opens.
   c) In the "Definition of segment" area, tap "Insert".
      Point 1 appears.
   d) Tap the X-cell and enter the X-value of the point.
   e) Tap the Y-cell and enter the Y-value of the point.
   f) To insert the next point:
      - Tap "Insert" to add a point before the selected point.
      - Tap "Append" to add a point at the end of the list.
   g) Set the X- and Y-values for this point.
   h) Repeat the last two steps until you define all points.
7. In the "Define Mask" tab, tap "Enable".
8. Adjust the horizontal and vertical units if necessary.

10.4.1.2 Modifying user masks

To change an existing mask definition, you can also use the graphical method on the touch screen, or the numerical way, or combine both.

With the graphical method, you can:

- Move, add, and delete segments
- Move and delete points

Adding points to an existing segment graphically is not possible.

With the numerical method, in the "Define Mask" tab, you have all modification possibilities. You can delete and add points and segments, change the coordinates, and also stretch a segment, or move it by adding an offset.

To add a mask segment on the touch screen

1. Tap a mask segment of the mask test that you want to complement.
2. Tap the "Masks" icon on the toolbar.

3. Tap the corner points of the new mask segment on the touch screen.
4. To finish the segment and mask definition, double-tap the last point.

To delete a mask segment on the touch screen

1. On the toolbar, tap the "Delete" icon.

2. Tap the mask segment you want to delete.

To delete a point on the touch screen

1. Tap the mask segment from which you want to delete a point.

   The selected segment is now in definition mode, shown with blue color.
2. On the toolbar, tap the "Delete" icon.

   The "Delete" overlay menu opens.
3. Tap the point you want to delete.

To move a segment on the touch screen

1. Drag&drop the segment to the new position.
2. Tap outside the mask to deselect the mask segment.

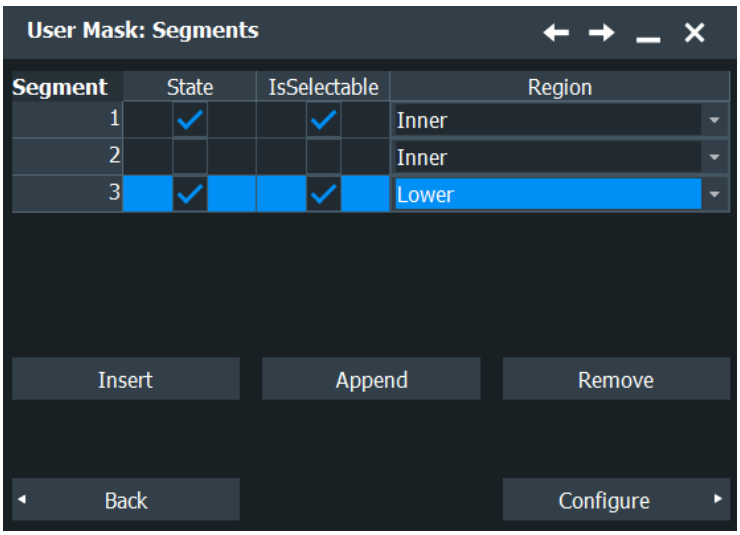
To move a point on the touch screen

1. Tap the mask segment that you want to change.
2. Drag&drop the point to the new position.
3. Tap outside the mask to deselect the mask segment.

To change the mask definition numerically

The settings mentioned here are described in detail in Chapter 10.2.2, "Mask definition", on page 398.

1. Open [App Cockpit] > "Analysis" tab > "Mask" > "Define Mask".
2. At the top, select the mask test for which you want to change the mask definition.
3. To add or delete a mask segment, tap "Setup mask".
   Tap the segment's row in the "User Mask: Segments" table and tap the required button below:
   - "Insert": to add a new segment before the selected segment.
   - "Append": to add a new segment at the end of the list.
   - "Remove": to delete the selected mask segment from the list.



4. To add, delete, or move a point of a segment:
   a) Select the segment in the "User Mask: Segments" table.
   b) Tap "Configure".
   c) Select the point in the "Definition of segment" table.
   d) To add or delete the selected point, use the buttons below the table.
      - "Insert": to add a new point before the selected point.
      - "Append": to add a new point at the end of the list.
      - "Remove": to delete the selected point from the list.

e) To move the selected point, change the X- and Y-values.

To rescale and move a mask segment

The settings mentioned here are described in detail in Chapter 10.2.2, "Mask definition", on page 398.

1. Open [App Cockpit] > "Analysis" tab > "Mask" > "Define Mask".
2. At the top, select the mask test for which you want to change the mask definition.
3. Tap "Setup mask" and select the required segment in the "User Mask: Segments" table.
4. Tap "Configure".
5. To stretch or compress the selected mask segment, enter the "Factor X" for horizontal scaling and the "Factor Y" for vertical scaling. The x-values and y-values of all points are multiplied with the corresponding factor. Factors >1 stretch the mask segment, while factors between 0 and 1 compress it. Negative values are possible and change the algebraic sign.
6. To move the selected mask segment, enter the "Offset X" for horizontal direction and the "Offset Y" for vertical direction. The specified offset is added to the corresponding values of all points.
7. Tap "Apply" to perform the scaling and/or move.

10.4.2 Setting up a mask test

In addition to the mask definition, the mask test contains further settings:

- The waveform to be tested
- The criteria for a failed test
- The actions to be taken if a test has failed or has been completed successfully

1. Open [App Cockpit] > "Analysis" tab > "Mask" > "Define Mask" tab.
2. Select the "Target" to be tested.
3. Select the "Define Test" tab.
4. Set the conditions for a failed test:
   a) " Fail condition": select if sample hits or the number of acquisitions with sample hits are considered.
   b) "Violation tolerance": number of tolerable sample hits or acquisition hits.

   A test has failed if the number of sample hits or acquisition hits exceeds the limit of violation tolerance hits.
5. Tap "Action on mask".
6. For each action, select when the action will be executed:
   - "On violation" if the mask test has failed
   - "On successful completion"

10.4.3 Configuring the mask and hit display

The display of masks and mask violation is the same for all mask tests .

The settings mentioned here are described in detail in Chapter 10.2.3, "Define Test", on page 405.

1. Open [App Cockpit] > "Analysis" tab > "Mask".
2. Select the "Display" tab.
3. Select "Show mask" to display the masks of all enabled mask tests on the screen.
4. Open "Menu" > "Settings" >"Appearance" dialog.
5. Select the "Mask" tab.
6. Define how the sample hits are displayed:
   a) Select "Show hits" to display the sample hits.
   b) Set the "Show for" time or "Show hits forever".
   c) Set the "Color" of the sample hits.
7. Define the color of the masks segments depending on the violation state:
   - Mask without violation
   - Mask with violation

- Mask with contact: this color shows that the edge of the mask segment was touched. In this case, the resolution is not sufficient to detect if the mask was really hit or not. Zoom into the concerned area to see the correct result.

10.4.4 Running a mask test

Before you can start a mask test, make sure that the mask setup is complete:

- The mask is defined, see Chapter 10.4.1.1, "Creating user masks", on page 411 and Chapter 10.4.1.2, "Modifying user masks", on page 412.
- The mask test is defined, see Chapter 10.4.2, "Setting up a mask test", on page 416
- The mask display is configured, see Chapter 10.4.3, "Configuring the mask and hit display", on page 416.

You can perform continuous testing or test a specified number of acquisitions.

1. Open [App Cockpit] > "Analysis" tab > "Mask" > "Define Mask" tab.
2. Select "Enable".

   If the acquisition is running, the test starts immediately.
3. If the acquisition is not running, press [Run Stop].

   The test starts and runs until you stop the acquisition or the stop action is executed if defined.

10.4.5 Saving and loading masks

Mask test definitions remain on the instrument until they are changed or deleted, or [Preset] is performed. If you want to keep a mask test, you can save and reload them.

To save a mask

1. Open [App Cockpit] > "Analysis" tab > "Mask" > "Define Mask" tab.
2. To save the mask file in the current directory, change the file name if needed, and tap "Save".
   You can use the automatic file name generation, see Chapter 4.8.1, "Autonaming", on page 104.
3. To select the directory and enter the file name, tap "Save As".

To load a mask

1. Tap "Open".
2. Select the file from the file selection dialog box.

10.4.6 Mask testing on history acquisitions

In the same way as for running acquisitions, you can set up and perform the mask testing also on history waveforms.

The requirements for mask testing on history waveforms are also the same:

- The mask is defined, see Chapter 10.4.1.1, "Creating user masks", on page 411 and Chapter 10.4.1.2, "Modifying user masks", on page 412.
- The mask test is defined, see Chapter 10.4.2, "Setting up a mask test", on page 416
- The mask display is configured, see Chapter 10.4.3, "Configuring the mask and hit display", on page 416.

1. Perform and finish the acquisition.
2. Press [History].
3. In the quick-access "History" dialog box, tap "Play".

   The mask testing is performed on the complete history memory, starting with the oldest acquisition. The state of the mask test is set to "Finished" when "Nx Single count" acquisitions are tested.

For details on history, see Chapter 7.4.1, "About history", on page 274.

11 Search functions

Search functions allow you to detect and analyze specific events in the acquired data quickly and simply. You can search in various waveforms for several events at once. The search area can be limited by a gate.

The events that can be searched for are basically the same events you can trigger on. Thus, the search parameters are defined in the same way as the trigger conditions. The results are displayed in a result box and optionally shown in a zoom window.

11.1 Overview: search definition and results

11.1.1 Search definition

You can define up to 8 different searches and let them run simultaneously. For each search, you define the criteria, the parameters of each criterion, the gate, and the result display.

The instrument keeps the settings until the next preset. If you save a user-defined preset, the search settings are included in the preset.

Each search is configured in a separate tab and contains:

- *Search control*
  If you enable a search and run an acquisition, the search is performed continuously on the acquired data until acquisition is stopped.
  If acquisition is stopped and you enable a search, the data of the last acquisition is searched.
  Enabling the search zoom window disables the search, stops a running acquisition, and displays the search results of the last acquisition in the zoom window.
- *Source*
  Waveform that is searched for one or more events. You can search in analog and digital signals, math or reference waveforms, and tracks. Furthermore, search in decoded data of serial buses is possible.
- *Search criteria and parameters*
  Various search criteria are available, depending on the source. Most parameters known from trigger event definition can also be configured as search conditions. Unlike triggering, you can configure several event types to be searched for simultaneously.
  If the source is an FFT spectrum, you can perform a frequency marker search by using the cursor measurement and defining the peak excursion. See Chapter 8.1.3.3, "Peak Search tab", on page 293.
- *Search gate*
  Searches can be performed on the entire waveform, or only on a defined area (gate). The gate can be coupled to an existing zoom.
  Gating is not available for searches on digital signals and serial buses.

- *Result presentation*
  For each search, you define how the search results are displayed: in a result table and/or in a search zoom window.

Remote commands:

- `SEARch:ADD` on page 1604
- `SEARch:REMove` on page 1605

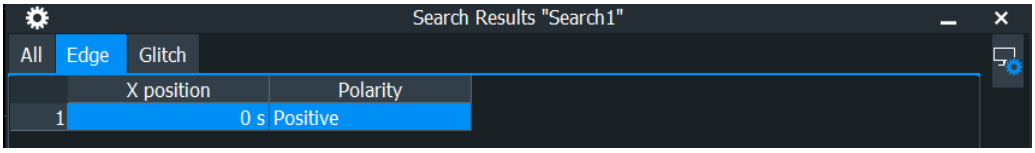
11.1.2 Search results

The results are displayed in a "Search Results" box and optionally in a zoom window.

Search Results box

The results of each search are tabulated in a "Search Results" box.

If you search for several event types in parallel, the results are presented in several tabs - one for each search event and one for the combined results. Each tab contains a table with the position and, if available, further parameters for each result. The tables row can be sorted, and you can define a maximum number of table entries in the "Result Presentation" dialog box. As with all result boxes, you can minimize it, display it like a diagram, and define the default position.



If "Auto clear" is enabled in the "Display" dialog box, the instrument displays the search results of the last acquisition. If "Auto clear" is disabled, the first result of each acquisition is listed until the maximum number of entries in the table is reached.

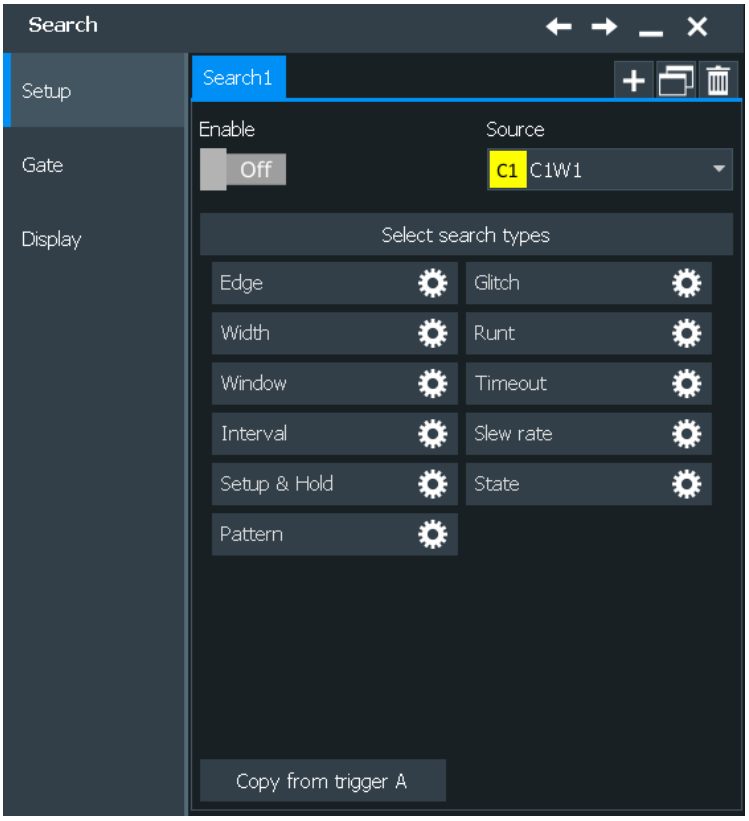
Remote commands for result query:

- `SEARch:RESult[:ALL]?` on page 1639

11.2 Search setup

Access: "Menu" > "Apps" > "Analysis" tab > "Search"

The search setup includes the source selection, the selection of search events (criteria), event-specific search conditions, and search control.



11.2.1 Search criteria

Access: "Menu" > "Apps" > "Analysis" tab > "Search" > "Setup" tab

Enable
If you enable a search and run an acquisition, the search is performed continuously on the acquired data until acquisition is stopped.

If acquisition is stopped and you enable a search, the data of the last acquisition is searched.

Remote command:
`SEARch:ONLine` on page 1605
`SEARch:ALL` on page 1605

Source
Defines the waveform to be searched. The source can be any analog and digital input signal, math or reference waveform, or track. While the instrument triggers only on real input signals, it can search also calculated and restored waveforms.

If the source is an FFT spectrum, you can perform a frequency marker search by using the cursor measurement and defining the peak excursion. See Chapter 8.1.3.3, "Peak Search tab", on page 293.

For some serial protocol options, search on a serial bus is available. For details, see the relevant chapters of the "Protocol Analysis" chapter.

Depending on the selected source, different search criteria are available.

Remote command:
SEARch:SOURce on page 1605

Edge, Glitch, Width, Runt, Window, Timeout, Interval, Slew rate, Setup & Hold, State, Pattern
Search criteria for analog and digital input signals, math and reference waveforms, and tracks. For searching on digital channels, only "Edge", "Width", "Timeout" and "Setup & Hold" criteria are available.

Tap the icon to include or exclude the search criteria in the next search. You can enable several event types for simultaneous search.

Remote command:
SEARch:TRIGger:EDGE[:STATe] on page 1606
SEARch:TRIGger:GLITch[:STATe] on page 1606
SEARch:TRIGger:WIDTh[:STATe] on page 1607
SEARch:TRIGger:RUNT[:STATe] on page 1606
SEARch:TRIGger:WINDow[:STATe] on page 1607
SEARch:TRIGger:TIMeout[:STATe] on page 1607
SEARch:TRIGger:INTerval[:STATe] on page 1606
SEARch:TRIGger:SLEWrate[:STATe] on page 1607
SEARch:TRIGger:SETHold[:STATe] on page 1606
SEARch:TRIGger:STATe[:STATe] on page 1607
SEARch:TRIGger:PATTern[:STATe] on page 1606

Copy from trigger A'
Copies the trigger type-specific settings from the A-trigger configuration to the search settings. The source itself is not copied.

Remote command:
SEARch:TRIGger:EDGE:ACOPy on page 1607
SEARch:TRIGger:GLITch:ACOPy on page 1607
SEARch:TRIGger:WINDow:ACOPy on page 1608
SEARch:TRIGger:WIDTh:ACOPy on page 1608
SEARch:TRIGger:RUNT:ACOPy on page 1607
SEARch:TRIGger:WINDow:ACOPy on page 1608
SEARch:TRIGger:TIMeout:ACOPy on page 1608
SEARch:TRIGger:INTerval:ACOPy on page 1607
SEARch:TRIGger:SLEWrate:ACOPy on page 1607
SEARch:TRIGger:SETHold:ACOPy on page 1607
SEARch:TRIGger:STATe:ACOPy on page 1608
SEARch:TRIGger:PATTern:ACOPy on page 1607

11.2.2 Search parameters

Most parameters available for trigger event definition can also be configured as search conditions. Each event type is defined in a separate subtab.

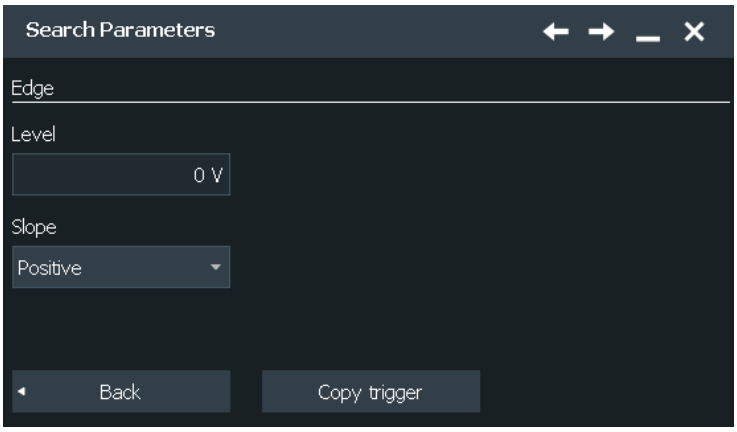
If the source is a spectrum, the instrument performs a frequency marker search.

For serial protocol options, search on a serial bus is available. These searches have protocol-specific search criteria. For details, see the relevant chapters of the "Protocol Analysis" chapter.

- Edge....423
- Glitch....424
- Width....424
- Runt....425
- Window....426
- Timeout....427
- Interval....428
- Slew rate....429
- Setup & Hold....430
- State....432
- Pattern....432

11.2.2.1 Edge

The edge search works the same way as the edge trigger.



Slope
Sets the edge type: rising edge ("Positive"), falling edge ("Negative"), or both.

Remote command:
`SEARch:TRIGger:EDGE:SLOPe` on page 1608

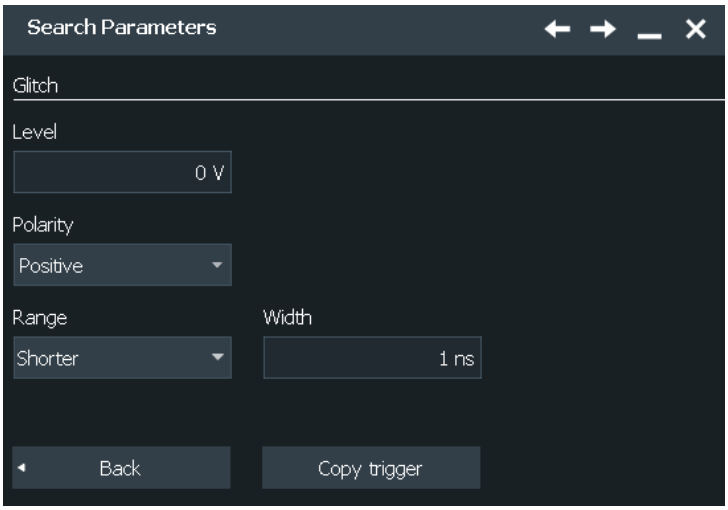
Level
Sets the voltage level for trigger search. The value is used for all search event types that require one trigger level. The search trigger level is search-specific, you can define different levels in different searches for the same event.

Remote command:
`SEARch:TRIGger:LEVel[:VALue]` on page 1607

11.2.2.2 Glitch

The glitch search works the same way as the glitch trigger. To apply the trigger settings to search, tap Copy from trigger A'.

The glitch search is not available if the search source is a digital channel.



Polarity, Range, Width
See trigger settings:

- "Range" on page 197
- "Width" on page 197
- "Polarity" on page 197

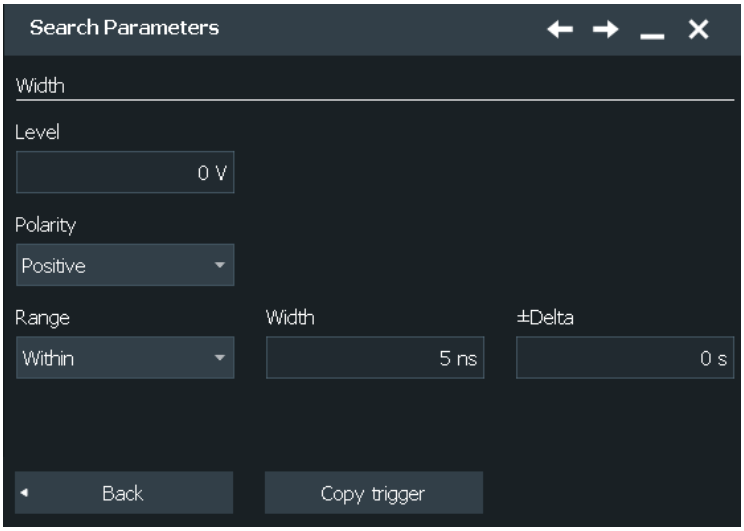
Remote command:
`SEARch:TRIGger:GLITch:POLarity` on page 1609
`SEARch:TRIGger:GLITch:RANGe` on page 1609
`SEARch:TRIGger:GLITch:WIDTh` on page 1609

Level
Sets the voltage level for trigger search. The value is used for all search event types that require one trigger level. The search trigger level is search-specific, you can define different levels in different searches for the same event.

Remote command:
`SEARch:TRIGger:LEVel[:VALue]` on page 1607

11.2.2.3 Width

The width search works the same way as the width trigger. To apply the trigger settings to search, tap Copy from trigger A'.



Polarity, Range, Width, ±Delta
See trigger settings:

- "Polarity" on page 198
  While the width trigger can only analyze positive *or* negative polarity, searching for a width is also possible for both polarities at the same time ("Either").
- "Range" on page 199
- "Width" on page 199
- "±Delta" on page 199

Remote command:
`SEARch:TRIGger:WIDTh:POLarity` on page 1617
`SEARch:TRIGger:WIDTh:RANGe` on page 1618
`SEARch:TRIGger:WIDTh:WIDTh` on page 1618
`SEARch:TRIGger:WIDTh:DELTa` on page 1617

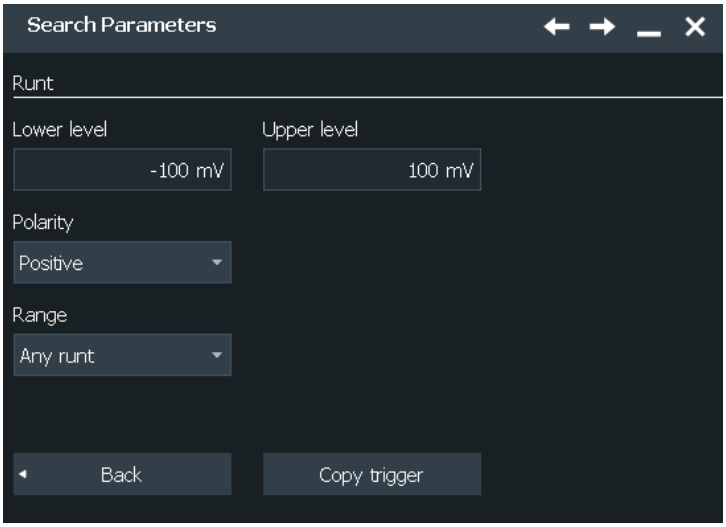
Level
Sets the voltage level for trigger search. The value is used for all search event types that require one trigger level. The search trigger level is search-specific, you can define different levels in different searches for the same event.

Remote command:
`SEARch:TRIGger:LEVel[:VALue]` on page 1607

11.2.2.4 Runt

The runt search settings are the same as the runt trigger settings. To apply the trigger settings to search, tap Copy from trigger A'.

The runt search is not available, if the search source is a digital channel.



Polarity, Range, Runt width, ±Delta
Time limit for the runt, see trigger settings:

- "Polarity" on page 197
- "Range" on page 200
- "Runt width" on page 201
- "±Delta" on page 201

Remote command:
`SEARch:TRIGger:RUNT:POLarity` on page 1612
`SEARch:TRIGger:RUNT:RANGe` on page 1612
`SEARch:TRIGger:RUNT:WIDTh` on page 1613
`SEARch:TRIGger:RUNT:DELTa` on page 1611

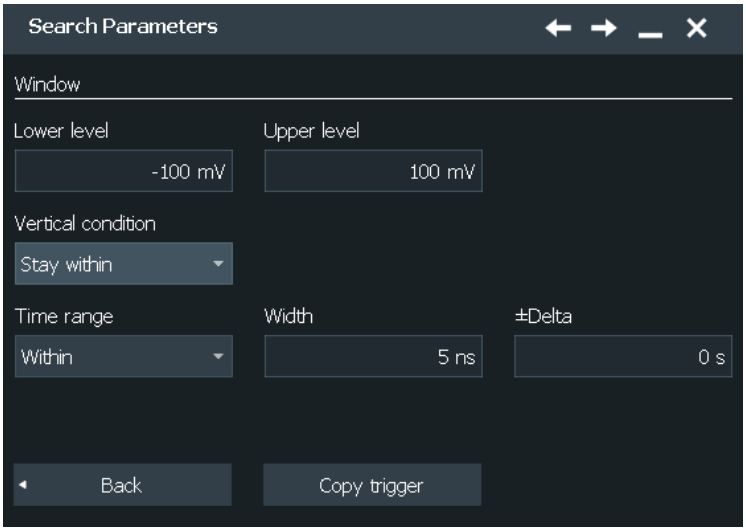
Upper level, Lower level
Set the upper and lower voltage thresholds. The amplitude of a runt crosses the first threshold twice in succession without crossing the second one.

Remote command:
`SEARch:TRIGger:LEVel:RUNT:UPPer` on page 1613
`SEARch:TRIGger:LEVel:RUNT:LOWer` on page 1613

11.2.2.5 Window

The window search settings are the same as the window trigger settings. This search type is not available if the search source is a digital channel. To apply the trigger settings to search, tap Copy from trigger A'.

The window search is not available if the search source is a digital channel.



Vertical condition
Defines the run of the signal relative to the window, see "Vertical condition" on page 202.

Remote command:
`SEARch:TRIGger:WINDow:RANGe` on page 1619

Time condition, Width, ±Delta
Set the time limit for the vertical condition, see:
- "Time condition" on page 202
- "Width" on page 202
- "±Delta" on page 203

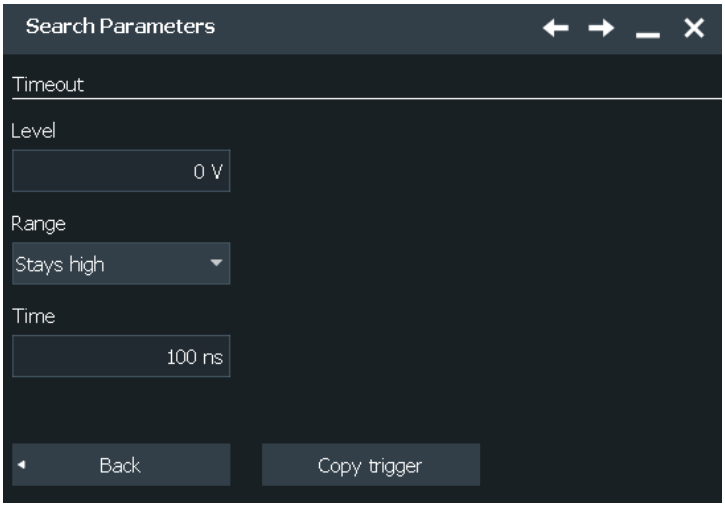
Remote command:
`SEARch:TRIGger:WINDow:TIMerange` on page 1620
`SEARch:TRIGger:WINDow:WIDTh` on page 1620
`SEARch:TRIGger:WINDow:DELTa` on page 1619

Upper level, Lower level
Set the upper and lower voltage thresholds. The amplitude of a runt crosses the first threshold twice in succession without crossing the second one.

Remote command:
`SEARch:TRIGger:LEVel:WINDow:UPPer` on page 1621
`SEARch:TRIGger:LEVel:WINDow:LOWer` on page 1621

11.2.2.6 Timeout

The timeout search settings are the same as the timeout trigger settings. To apply the trigger settings to search, tap Copy from trigger A'.



Range, Time
Set the timeout condition, see:
- "Range" on page 203
- "Time" on page 204

Remote command:
`SEARch:TRIGger:TIMeout:RANGe` on page 1616
`SEARch:TRIGger:TIMeout:TIME` on page 1616

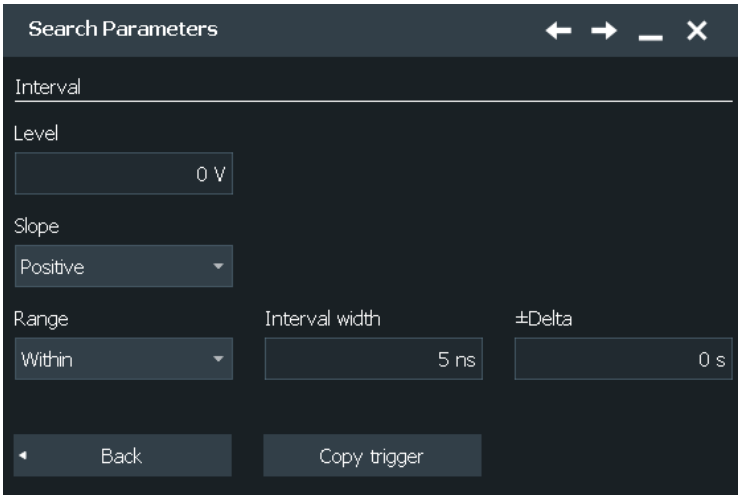
Level
Sets the voltage level for trigger search. The value is used for all search event types that require one trigger level. The search trigger level is search-specific, you can define different levels in different searches for the same event.

Remote command:
`SEARch:TRIGger:LEVel[:VALue]` on page 1607

11.2.2.7 Interval

The interval search settings are the same as the interval trigger settings. To apply the trigger settings to search, tap Copy from trigger A'.

The interval search is not available if the search source is a digital channel.



Slope, Range, Interv. width, ±Delta
Set the interval condition, see:

- "Slope" on page 204
  While the interval trigger can only analyze rising or falling edges, searching for a width is possible for both edges at the same time ("Either").
- "Range" on page 204
- "Interv. width" on page 205
- "±Delta" on page 205

Remote command:
`SEARch:TRIGger:INTerval:SLOPe` on page 1610
`SEARch:TRIGger:INTerval:RANGe` on page 1610
`SEARch:TRIGger:INTerval:WIDTh` on page 1611
`SEARch:TRIGger:INTerval:DELTa` on page 1610

Level
Sets the voltage level for trigger search. The value is used for all search event types that require one trigger level. The search trigger level is search-specific, you can define different levels in different searches for the same event.

Remote command:
`SEARch:TRIGger:LEVel[:VALue]` on page 1607

11.2.2.8 Slew rate

The slew rate search settings are the same as the slew rate trigger settings. To apply the trigger settings to search, tap Copy from trigger A'.

The slew rate search is not available if the search source is a digital channel.

Polarity, Range, Slew rate, ±Delta

- "Slope" on page 196
- "Range" on page 206
- "Slew rate" on page 207
- "±Delta" on page 207

Remote command:
`SEARch:TRIGger:SLEWrate:SLOPe` on page 1615
`SEARch:TRIGger:SLEWrate:RANGe` on page 1614
`SEARch:TRIGger:SLEWrate:TIME` on page 1615
`SEARch:TRIGger:SLEWrate:DELTa` on page 1614

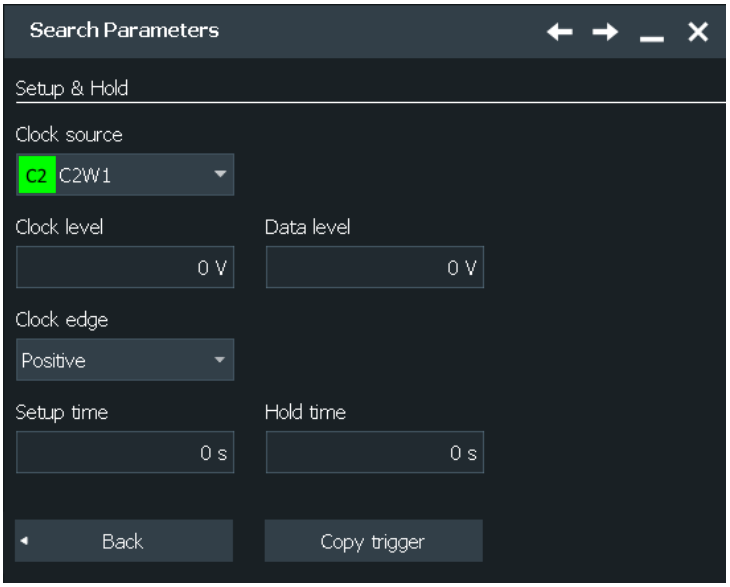
Upper level, Lower level

Set the upper and lower voltage thresholds. When the signal crosses a level, the slew rate measurement starts or stops depending on the selected slope.

Remote command:
`SEARch:TRIGger:LEVel:TRANsition:UPPer` on page 1615
`SEARch:TRIGger:LEVel:TRANsition:LOWer` on page 1615

11.2.2.9 Setup & Hold

The Setup & Hold search settings are the same as the Setup & Hold trigger settings. To apply the trigger settings to search, tap Copy from trigger A'.



Clock source, Clock edge, Clock level
Set the clock settings. Both "Clock level" and "Clock edge" define the starting point for calculation of the setup and hold time.

Remote command:
`SEARch:TRIGger:SETHold:CSOurce` on page 1622
`SEARch:TRIGger:SETHold:CEDGe` on page 1621
`SEARch:TRIGger:SETHold:CLEVel` on page 1622

Data level
Sets the voltage level for the data signal. At this level, the setup and hold time are measured.

Remote command:
`SEARch:TRIGger:LEVel[:VALue]` on page 1607

Level
Sets the voltage level for trigger search. The value is used for all search event types that require one trigger level. The search trigger level is search-specific, you can define different levels in different searches for the same event.

Remote command:
`SEARch:TRIGger:LEVel[:VALue]` on page 1607

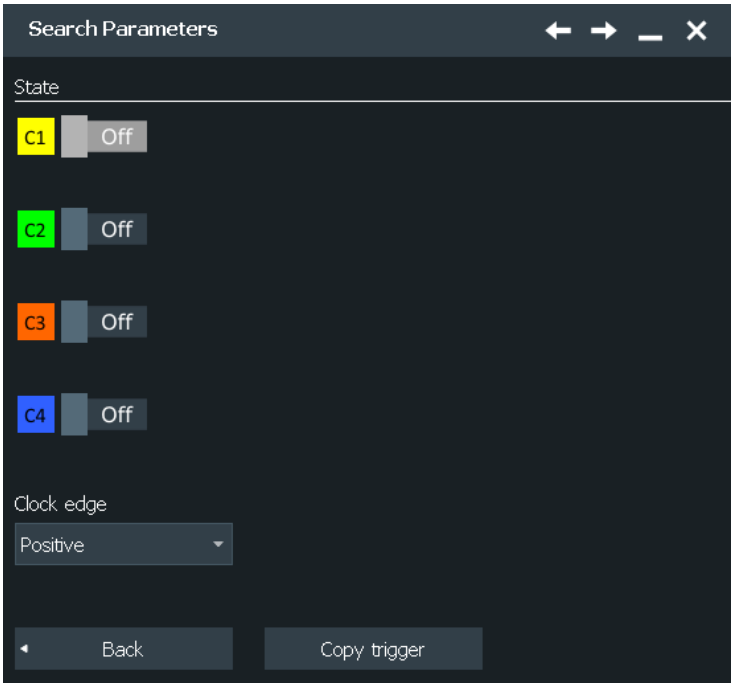
Setup time, Hold time
Sets the minimum time **before** (Setup) and **after** (Hold) the clock edge while the data signal must stay steady above or below the data level.

See also: "Setup time" on page 209 and "Hold time" on page 209.

Remote command:
`SEARch:TRIGger:SETHold:STIMe` on page 1623
`SEARch:TRIGger:SETHold:HTIMe` on page 1622

11.2.2.10 State

The state search is a qualified edge search. The state search is only available for analog channel sources (Ch1 to Ch4).



Clock source, Clock edge
Define the clock settings. The clock signal is the waveform to be searched.

Remote command:
`SEARch:TRIGger:STATe:CSOurce` on page 1628
`SEARch:TRIGger:STATe:CEDGe` on page 1628
`SEARch:TRIGger:STATe:CLEVel` on page 1628

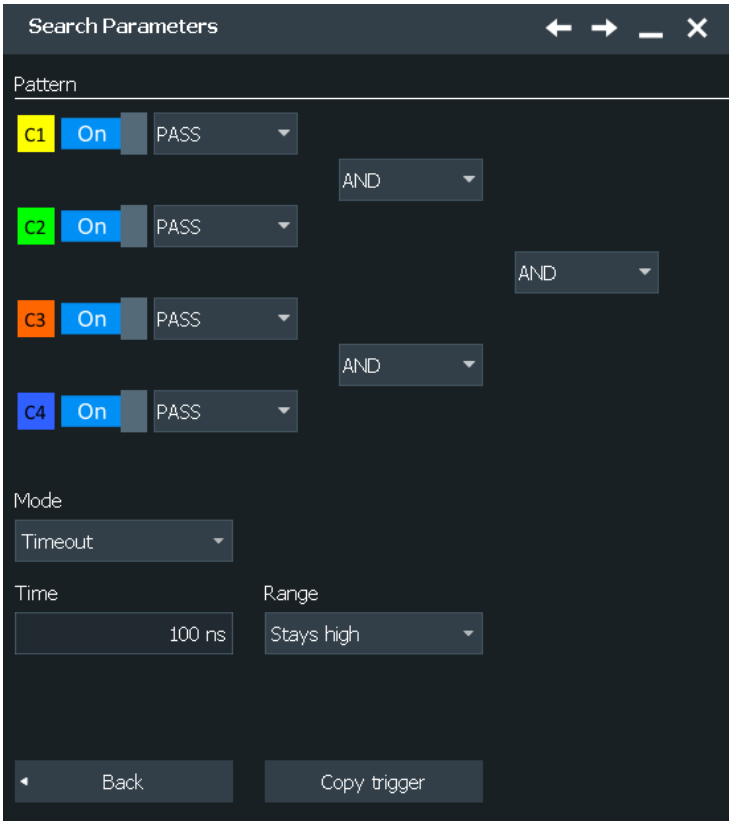
State pattern
State settings are the same as for the state trigger.

See also "Pattern" on page 211.

Remote command:
`SEARch:TRIGger:STATe:A[:ENABle]` on page 1629
`SEARch:TRIGger:STATe:A:LOGic` on page 1629
`SEARch:TRIGger:STATe:AB:LOGic` on page 1630
`SEARch:TRIGger:STATe:ABCD:LOGic` on page 1630

11.2.2.11 Pattern

The pattern search combines a logical combination of the input channels with a timing condition. The pattern search is only available for analog channel sources (Ch1 to Ch4).



Pattern

Pattern search settings are the same as for the pattern trigger.

See also "Pattern" on page 211.

Remote command:

`SEARch:TRIGger:PATTern:A[:ENABle]` on page 1624
`SEARch:TRIGger:PATTern:A:LOGic` on page 1624
`SEARch:TRIGger:PATTern:AB:LOGic` on page 1625
`SEARch:TRIGger:PATTern:ABCD:LOGic` on page 1625

Timing condition: Mode, Range, Time, Width, ±Delta

Additional time limitation to the pattern, see "Additional settings: Timing" on page 212.

Remote command:

`SEARch:TRIGger:PATTern:MODE` on page 1625
`SEARch:TRIGger:PATTern:TIMeout:MODE` on page 1626
`SEARch:TRIGger:PATTern:TIMeout[:TIME]` on page 1626
`SEARch:TRIGger:PATTern:WIDTh:RANGe` on page 1626
`SEARch:TRIGger:PATTern:WIDTh[:WIDTh]` on page 1627
`SEARch:TRIGger:PATTern:WIDTh:DELTa` on page 1627

11.2.3 Frequency marker search

When you start a search on a spectrum, a frequency marker search is performed to detect peaks in a spectrum. You can define which peaks the instrument finds by defining the noise reject settings.

Threshold
See "Threshold" on page 294.

Peak excursion
See "Peak excursion" on page 295.

11.2.4 Configuring the search setup

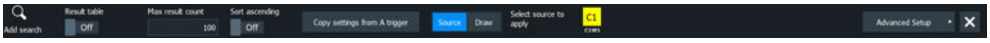
There are several ways to create a search:

- Creating a simple default search using the toolbar icon. This method is not available for search on serial buses.
- Setting up a search using the dialog box.

To perform a simple search

1. If more than one waveform is in the diagram, select the waveform to be searched for by tapping it in the diagram.
2. Select the "Search" icon on the toolbar.

   The overlay menu opens.

3. Tap the diagram with the waveform to be searched, or drag a rectangle on the diagram to define the search area.
4. Alternatively, use the settings provided in the overlay menu.

   The default edge search is configured as "Search<x>" and performed. The "Search Results" box is displayed.

To create a user-defined search

1. Open the search dialog: "Menu" > "Apps" > "Analysis" tab > "Search".
2. There are two ways to create a search:
   - If you want to create a new, unconfigured search, tap the "Add" icon.

   - If you want to create a new search based on an existing one, tap the "Copy" icon.

3. Enter a name for the search using the on-screen keyboard.

To configure a user-defined search

1. Open the search dialog: "Menu" > "Apps" > "Analysis" tab > "Search".
2. Select the "Setup" tab and the search that you want to configure.
3. Select the "Source" on which you want to perform the search.
4. Select the events that you want to include in the search.
5. Define the settings of the first search event.
   To use the same conditions as defined in the trigger configuration of the A-event, tap "Copy from trigger A'". The selected trigger settings are applied to the search settings.
6. Repeat the previous steps to define further events for the same search.
7. To perform the search only on a part of the waveform, configure the gate in the "Gate" tab as described in Chapter 11.3.2, "Defining the search gate", on page 437.

11.3 Search gate

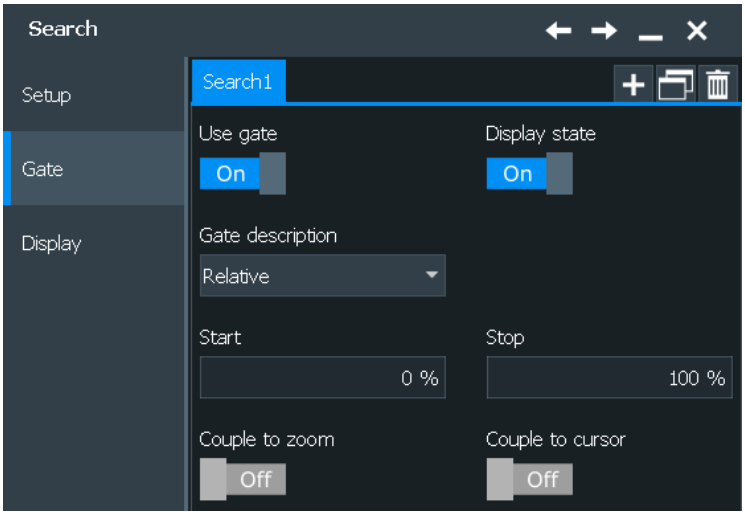
The gate defines the search area within the source waveform. You can use absolute or relative values to define the gate, or couple it to a previously defined zoom area.

11.3.1 Gate settings

Access: "Menu" > "Apps" > "Analysis" tab > "Search" > "Gate" tab

The search gate settings are identical to those for gate areas for measurements or FFT analysis.

Gating is not available if the search source is a digital channel or a serial bus.



Use gate
Enables the gate settings and shows the gate. Search is only performed on the defined gate area of the source waveform.

Remote command:
`SEARch:GATE[:STATe]` on page 1631

Display state
Displays the gate area in the source diagram.

Remote command:
`SEARch:GATE:SHOW` on page 1632

Gate description
Defines whether the gate settings are configured using absolute or relative values.

"Absolute" The gate is defined by absolute start and stop values.

"Relative" The gate's start and stop values are defined by a percentage of the value range.

Remote command:
`CALCulate:MATH<m>:FFT:GATE:MODE` on page 1573
`MEASurement<m>:GATE:MODE` on page 1547
`SEARch:GATE:MODE` on page 1632

(Relative) Start
Defines the starting value for the gate.

Remote command:
`CALCulate:MATH<m>:FFT:GATE:ABSolute:STARt` on page 1573
`CALCulate:MATH<m>:FFT:GATE:RELative:STARt` on page 1574
`MEASurement<m>:GATE:ABSolute:STARt` on page 1547
`MEASurement<m>:GATE:RELative:STARt` on page 1547
`SEARch:GATE:ABSolute:STARt` on page 1632
`SEARch:GATE:RELative:STARt` on page 1633

(Relative) Stop
Defines the end value for the gate.

Remote command:
`CALCulate:MATH<m>:FFT:GATE:ABSolute:STOP` on page 1573
`CALCulate:MATH<m>:FFT:GATE:RELative:STOP` on page 1574
`MEASurement<m>:GATE:ABSolute:STOP` on page 1547
`MEASurement<m>:GATE:RELative:STOP` on page 1547
`SEARch:GATE:ABSolute:STOP` on page 1632
`SEARch:GATE:RELative:STOP` on page 1633

Zoom
Zoom coupling is available if a zoom is defined. As long as "Zoom coupling" is enabled, the gate area is defined identically to the zoom area - if you change the zoom, the gate changes as well.

If several zoom diagrams are defined, select the zoom diagram to be used for gating. The "Start" and "Stop" values of the gate are adjusted accordingly.

Zoom coupling can be set for measurement gates, FFT gates, and search gates. The zoom must be defined on the diagram that contains the signal source of the measurement, FFT, or search.

Remote command:
`MEASurement<m>:GATE:ZCOupling` on page 1548
`MEASurement<m>:GATE:ZDIagram` on page 1549
`CALCulate:MATH<m>:FFT:GATE:ZCOupling` on page 1574
`SEARch:GATE:ZCOupling` on page 1633
`SEARch:GATE:ZDIagram` on page 1634

Cursor
If enabled, the gate area is defined by the cursor lines of an active cursor measurement.

If several cursor measurements are enabled, select the cursor set to be used for gating. The "Start" and "Stop" values of the gate are adjusted to the values of the cursor line positions. The measurement is limited to the part of the waveform between the cursor lines.

Remote command:
`MEASurement<m>:GATE:CCOupling` on page 1548
`MEASurement<m>:GATE:CURSor` on page 1548

11.3.2 Defining the search gate

If you create a search using the "Search" toolbar icon, you can directly define the gate by dragging a rectangle on the diagram. Otherwise, you define the gate in the "Gate" tab of the "Search" dialog box.

1. Open the "Menu" > "Apps" > "Analysis" tab > "Search" > "Gate" tab.
2. Select the search for which you want to define the gate.

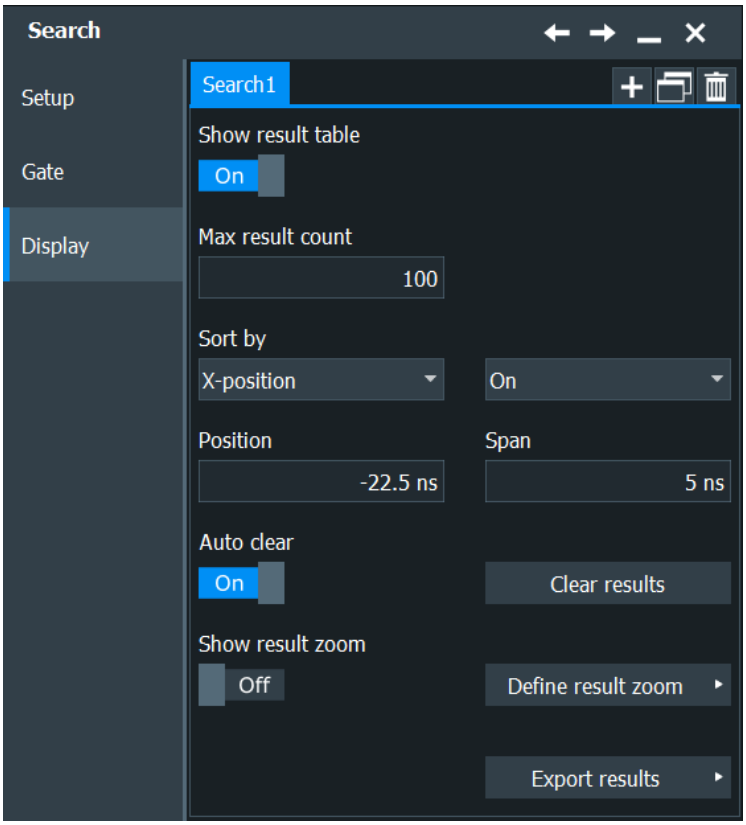
3. Use one of the following methods:
   a) Set the absolute or relative "Gate description" . Enter the start and stop values of the gate area.
   b) If a zoom area has already been defined for the waveform, couple the gate area to the zoom area by selecting the "Couple to zoom" option. If several zoom diagrams are defined, select the zoom diagram you want to use for gating.
4. Tap "Use gate" to enable the gate.
5. Optionally, tap "Display state" to display the gate area in the diagram.

11.4 Search display presentation

Search results are displayed in a table in the "Search Results" box. In addition, a zoom window for a selected search result can be displayed so that you can analyze the result in more detail.

11.4.1 Search display settings

Access: "Menu" > "Apps" > "Analysis" tab > "Search" > "Display"



The following settings configure the layout of the result table in the "Search Results" box and the size and position of the search zoom window. The result tables can be sorted by x-position or value. You can define a maximum number of table entries.

Show result table
Displays or hides the search result table.

Remote command:
`SEARch:RESult:SHOW` on page 1638

Max result count
Defines the maximum number of entries in the search result table.

Remote command:
`SEARch:RESult:LIMit` on page 1638

Sort by
Sorts the search results by x-value position or value of the result.

Remote command:
`SEARch:RESult:SORT[:MODE]` on page 1639

Sort ascending
By default, the results are listed in descending order, i.e. the largest value at the top. To change the sorting direction, enable "Sort ascending".

Remote command:
`SEARch:RESult:SORT:ASCending` on page 1638

Clear results
Clears the search results once and starts a new search.

Remote command:
`SEARch:CLEar` on page 1604

Auto clear
If "Auto clear" is enabled, the instrument displays the search results of the last acquisition.

If "Auto clear" is disabled, the first result of each acquisition is listed until the maximum number of entries in the table is reached.

Show result zoom
If enabled, a zoom window is displayed for the currently selected search result. The zoom area is indicated in the diagram that displays the source waveform of the search.

Remote command:
`SEARch:RESDiagram:SHOW` on page 1636

Define result zoom
Opens a dialog to define the reult zoom display settings.

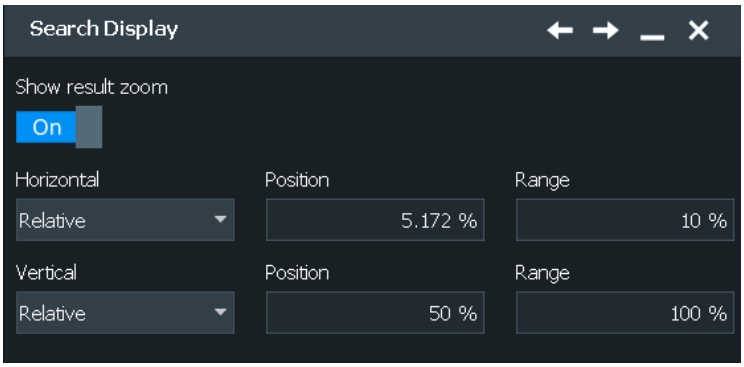
See Chapter 11.4.2, "Display zoom settings", on page 440.

Export results
Opens a dialog to define the export reults.

See Chapter 12.2.4, "Results", on page 467.

11.4.2 Display zoom settings

You can display a zoom window for the currently selected search result. The zoom area is indicated in the diagram that displays the source waveform of the search.



Show result zoom
If enabled, a zoom window is displayed for the currently selected search result. The zoom area is indicated in the diagram that displays the source waveform of the search.

Remote command:
`SEARch:RESDiagram:SHOW` on page 1636

Y mode
Defines whether absolute or relative values are used to specify the y-axis values.

Remote command:
`LAYout:ZOOM:VERTical:MODE` on page 1442
`SEARch:RESDiagram:VERT:MODE` on page 1637

Y position
Defines the y-value of the centerpoint of the zoom area.

Remote command:
`LAYout:ZOOM:VERTical:ABSolute:POSition` on page 1443
`LAYout:ZOOM:VERTical:RELative:POSition` on page 1444
`SEARch:RESDiagram:VERT:ABSolute:POSition` on page 1636
`SEARch:RESDiagram:VERT:RELative:POSition` on page 1637

Y range
Defines the height of the zoom area.

Remote command:
`LAYout:ZOOM:VERTical:RELative:SPAN` on page 1445
`LAYout:ZOOM:VERTical:ABSolute:SPAN` on page 1443

SEARch:RESDiagram:VERT:ABSolute:SPAN on page 1636
SEARch:RESDiagram:VERT:RELative:SPAN on page 1637

X mode
Defines whether absolute or relative values are used to specify the x-axis values.

Remote command:
LAYout:ZOOM:HORZ:MODE on page 1439
SEARch:RESDiagram:HORZ:MODE on page 1635

X position
Defines the x-value of the centerpoint of the zoom area.

Remote command:
LAYout:ZOOM:HORZ:ABSolute:POSition on page 1439
LAYout:ZOOM:HORZ:RELative:POSition on page 1441
SEARch:RESDiagram:HORZ:ABSolute:POSition on page 1634
SEARch:RESDiagram:HORZ:RELative:POSition on page 1635

X range
Defines the width of the zoom area.

Remote command:
LAYout:ZOOM:HORZ:ABSolute:SPAN on page 1440
LAYout:ZOOM:HORZ:RELative:SPAN on page 1441
SEARch:RESDiagram:HORZ:ABSolute:SPAN on page 1635
SEARch:RESDiagram:HORZ:RELative:SPAN on page 1636

11.4.3 Configuring the search results presentation

Initially, the "Search Results" box is displayed in front of the other diagrams or as result icon on the signal bar. This depends on the default setting in the "Diagram Layout" tab. Alternatively, you can display it in its own area on the screen, like any other diagram.

For details, see Chapter 3.3.7, "Displaying results", on page 58.

To configure the result tables

1. Open the search dialog: "Menu" > "Apps" > "Analysis" tab > "Search".
2. Select the tab for the search you want to configure.
3. Select the "Display" tab.
4. Select "Show result table" to display the "Search Results" box.
5. Select the sort mode of the result table.
6. By default, the results are listed in descending order, i.e. the largest value at the top. To change the sorting direction, enable "Sort ascending".
7. Define a maximum number of results to be displayed in the result table in the "Max. result count" field.

To display search zoom windows

1. In the "Search > Display" tab, enable "Show result zoom".

   This stops a running search and a running acquisition.
   The zoom area is indicated in the diagram that displays the source waveform of the search. The zoom window is displayed for the first result that was found.

2. If you need to adjust the search zoom area, you can drag the area or their edges on the screen. You can also enter the limits of the search zoom window in the "Search > Display > Define result zoom".
   Be aware, that the zoom window size is valid for all results of a search definition. If you change the settings drastically for one result, they may not be correct for the next search result you switch to.

   See also:
   - Chapter 7.1.3, "Zooming for details", on page 245

12 Data and file management

This chapter describes how to manage instrument settings, and measurement results like waveform data, numeric results and screenshots.

The "Save/Recall" dialog provides functions for saving and restoring data on the instrument. A naming pattern is available and can be adjusted to simplify a clear data storage.

The effect of the [Camera] key can be configured to save screenshots or reports.

- Save and recall user settings....443
- Save and recall waveform data and results....452
- Screenshots....475
- Reports....480
- File selection dialog....482

12.1 Save and recall user settings

To repeat measurements at different times or perform similar measurements with different test data, it is useful to save the used instrument settings and load them again later. Furthermore, it can be helpful to refer to the instrument settings of a particular measurement when analyzing the results. Therefore, you can easily save the instrument settings of a measurement. In addition to the measurement-related settings, user-specific display settings and active reference waveforms can also be saved and loaded.

Access: "Menu" key > "Save/Recall" key > "Save" tab > "User settings"

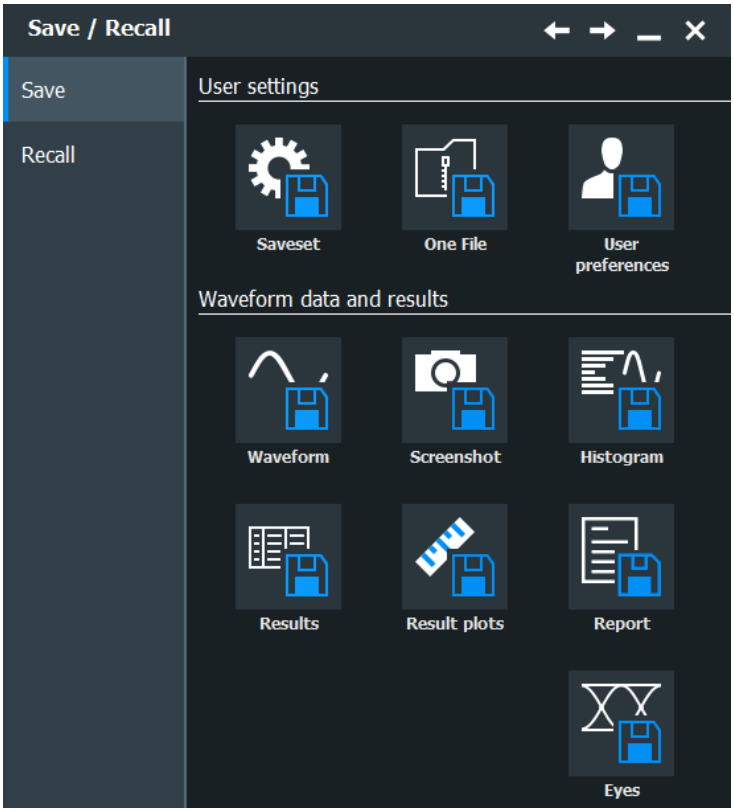


Figure 12-1: Save tab

Access to load instrument settings: "Menu" key > "Save/Recall" key > "Recall" tab > "User settings"

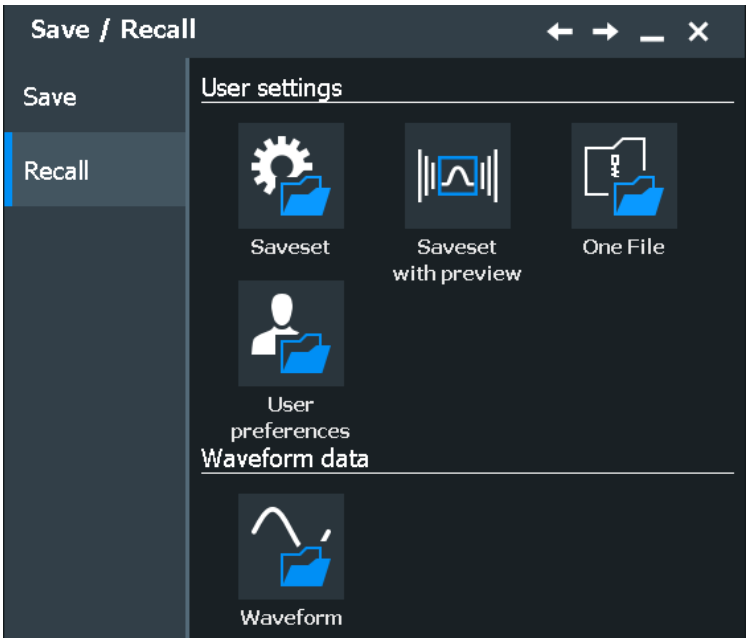


Figure 12-2: Recall tab

The R&S RTO6 provides the following types of saving and restoring settings:

- **Savesets** contain the complete instrument and measurement configuration except for user-specific display settings.
- **One File** contains the saveset and active reference waveform files in a ZIP file.
- **User preferences** contain the user specific display settings.
- **User-defined presets** contain the complete instrument setup including display settings, except for transparency and intensity. These settings can be restored by pressing the [Preset] key. See Chapter 4.6, "Preset setup", on page 98.

- Savesets 445
- One File 446
- User preferences 448
- Saveset with preview function 449
- Saving and loading settings 450

12.1.1 Savesets

Savesets contain the complete instrument and measurement configuration including a screenshot of the current display, but except for user-specific display settings stored as user preferences. You can save an unlimited number of setting files.

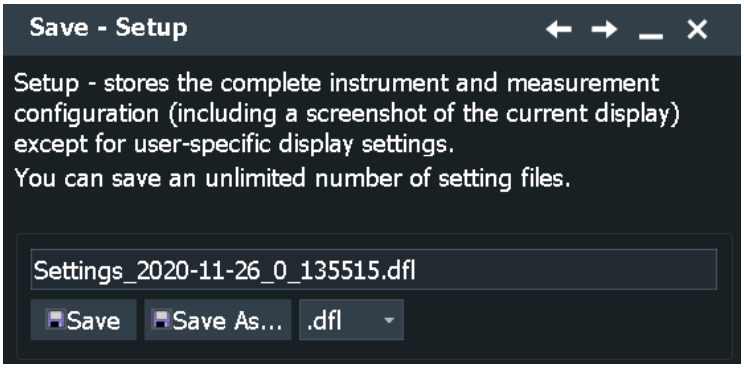
The waveform generator/pattern generator and the pulse source state is saved as "Off".

To load a saveset quickly, you can use the "Load saveset" function on the toolbar. A graphical preview helps you to find the required settings file.

If you need to store the instrument settings often, you can add the " Save settings" icon to the toolbar and use the icon to store the saveset file.

12.1.1.1 Save saveset settings

Access: "Menu" key > "Save/Recall" key > "Save" tab > "Savesets"



Save to file
Enter the file name to save the setting data to, and select the file format with the format button on the right. Double-tap the file name to open the file selection dialog box. See also: Chapter 12.5, "File selection dialog", on page 482.

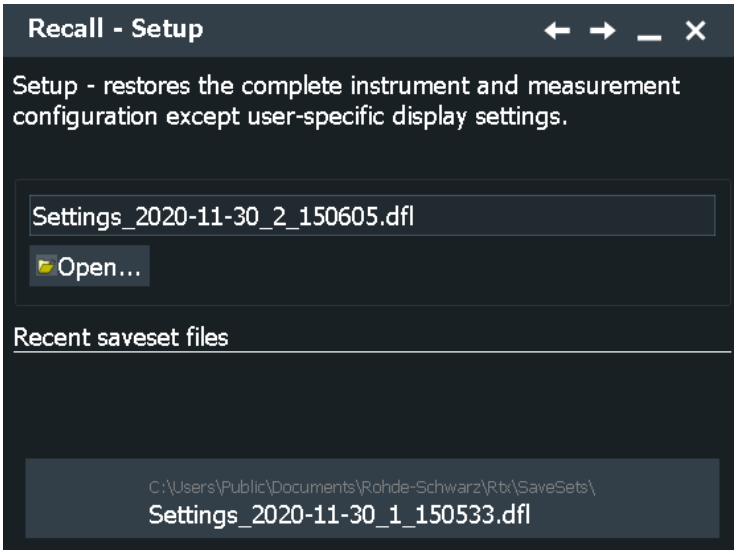
By default, saveset file names have the prefix "Settings_".

"Save"	Saves the data to the selected file.
"Save As..."	Opens the file selection dialog box and saves the data to the selected file.
".dfl/.xml"	Selects the file format.

Remote command:
`MMEMory:SAV` on page 1646

12.1.1.2 Load saveset settings

Access: "Menu" key > "Save/Recall" key > "Recall" tab > "Savesets"



Load from file
Enter the file name to load the setting data from, and select the file format with the format button on the right. Double-tap the file name to open the file selection dialog box. See also: Chapter 12.5, "File selection dialog", on page 482.

By default, saveset file names have the prefix "Settings_".

"Open"	Opens a file selection dialog box and loads the selected file.

Remote command:
`MMEMory:RCL` on page 1646

12.1.2 One File

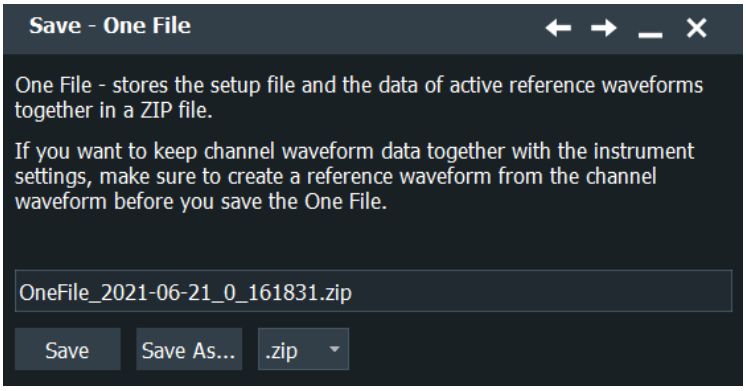
The One File function stores the savesets file and the data of active reference waveforms together in a ZIP file. If you want to keep channel waveform data together with

the instrument settings, make sure to create a reference waveform from the channel waveform before you save the One File.

The waveform generator/pattern generator and the pulse source state is saved as "Off".

12.1.2.1 Save One File settings

Access: "Menu" key > "Save/Recall" key > "Save" tab > "One File"



Save to file
Enter the file name to save the One File data to, and select the file format with the format button on the right. Double-tap the file name to open the file selection dialog box. See also: Chapter 12.5, "File selection dialog", on page 482.

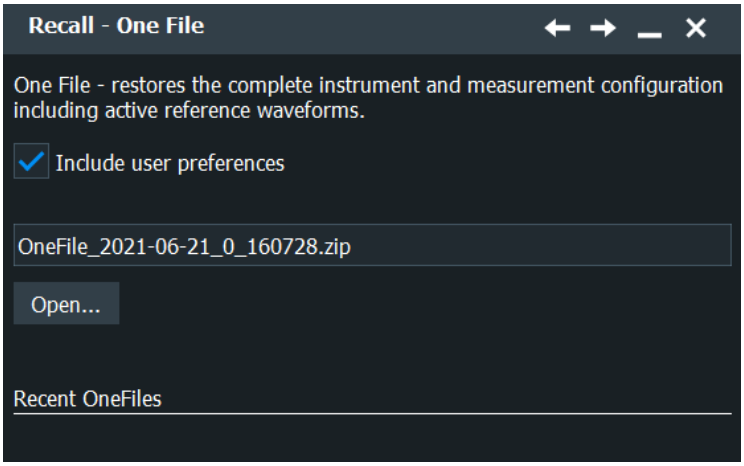
By default, one file names have the prefix "OneFile_".

"Save"	Saves the data to the selected file.
"Save As..."	Opens the file selection dialog box and saves the data to the selected file.
".zip"	Shows the file format.

Remote command:
`SAVeset:ONEFile:NAME` on page 1648
`SAVeset:ONEFile:SAVE` on page 1648

12.1.2.2 Load One File settings

Access: "Menu" key > "Save/Recall" key > "Recall" tab > "One File"



Load from file
Select "Open" to open the file selection dialog box, and to select the required file. See also: Chapter 12.5, "File selection dialog", on page 482.

By default, one file names have the prefix "OneFile_".

Remote command:
`SAVeset:ONEFile:NAME` on page 1648
`SAVeset:ONEFile:OPEN` on page 1648

Include user preferences
If enabled, the user preferences settings are also restored together with the saveset and reference waveform. User preferences settings include the user-specific display settings like diagram layout, toolbar, intensity and transparency settings.

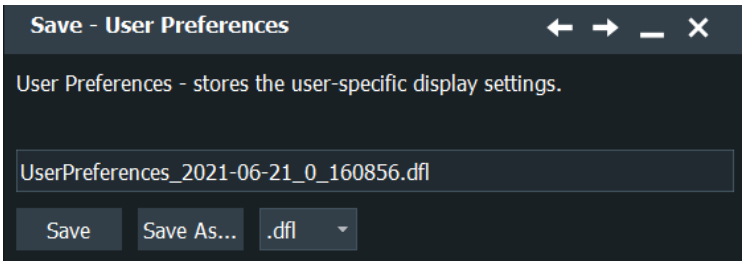
Remote command:
`SAVeset:ONEFile:USERpref` on page 1648

12.1.3 User preferences

User preferences contain user-specific display settings like diagram layout, toolbar, intensity and transparency settings.

12.1.3.1 Save user preferences settings

Access: "Menu" key > "Save/Recall" key > "Save" tab > "User Preferences"



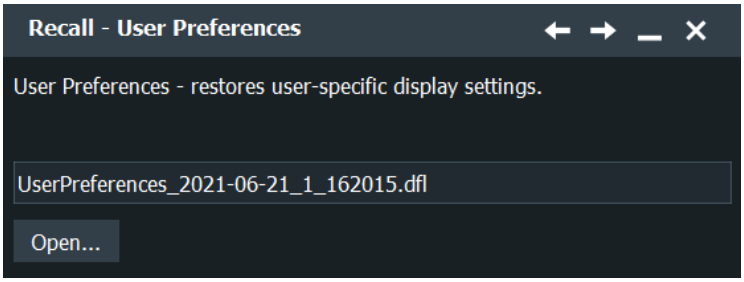
Save to file
The file name to save the data to.

By default, user preference file names have the prefix "UserPreferences_".

"Save"	Saves the data to the selected file.
"Save As..."	Opens the file selection dialog box and saves the data to the selected file.
".zip"	Shows the file format.

12.1.3.2 Load user preference settings

Access: "Menu" key > "Save/Recall" key > "Recall" tab > "User Preferences"



Load from file
Enter the file name to load the setting data from, and select the file format with the format button on the right. Double-tap the file name to open the file selection dialog box. See also: Chapter 12.5, "File selection dialog", on page 482.

By default, user preference file names have the prefix "UserPreferences_".

"Open"	Opens a file selection dialog box and loads the selected file.

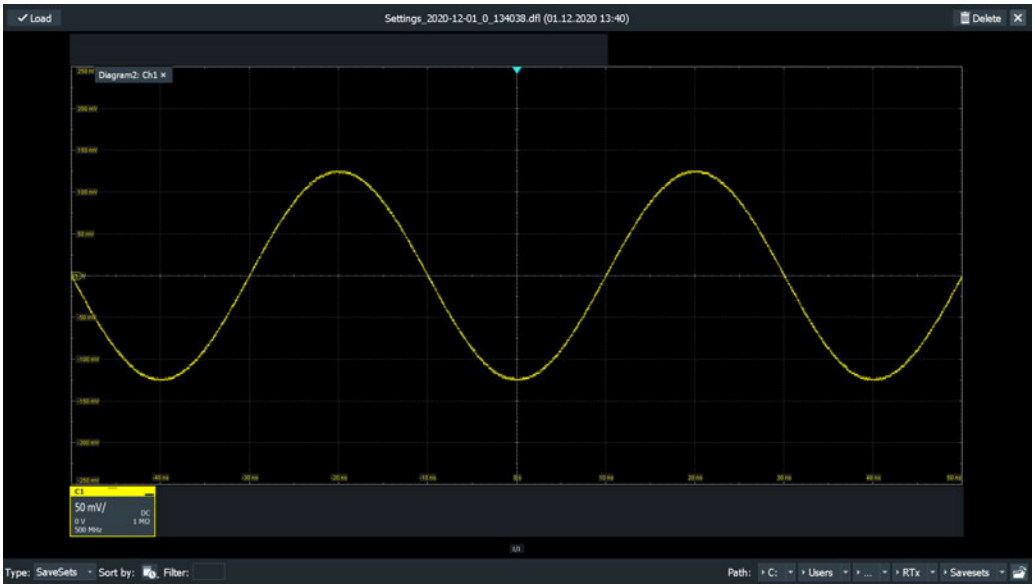
Remote command:
`MMEMory:RCL` on page 1646

12.1.4 Saveset with preview function

Access: "Menu" > "Save/Recall" > "Recall" tab > "Saveset with preview"

The graphical preview helps you to find the required settings file or a reference waveform.

The "Saveset with preview" function is also available on the toolbar, see Chapter 3.3.6.3, "Toolbar functions", on page 53.



1. On the toolbar, tap the "Saveset with preview " icon.

   A window opens and shows the screenshot of the first measurement configuration that is stored in the default directory.

2. Select if you want to display a "SaveSets" file or a reference waveform ("Ref Wfm").

3. Find the required file using one of the following methods:
   - Tap the "Next" icon on the right or the "Previous" icon on the left to scroll the savesets of the directory.
     The file name is shown on the top, and the screenshot helps to identify the settings.
   - If the saveset was stored in another directory, use the path buttons at the bottom, or tap "Open" to open the required directory.
   - Tap "Sort by" to sort the files according to the "Last modified" date or the "Filename".
   - Specify a "Filter" name to show only the files containing the filter string.

4. Tap "Load" in the upper left corner to recall the settings of the selected file.

12.1.5 Saving and loading settings

Settings can be stored in a file with user-defined name and location, or in a quick saveset. The settings in a saveset can be saved and retrieved quickly at the touch of a button, so savesets are ideal for frequently used measurements.

For details on save/recall instrument settings and associated remote commands, see Chapter 12.1.1, "Savesets", on page 445.

To save settings to a saveset file

Alternatively, you can add the " Save settings" icon to the toolbar and use the icon to store the saveset file to the folder and file specified in the "Settings" tab.

See also Chapter 3.3.6.2, "Configuring the toolbar", on page 53.

1. Tap "Menu" and select "Save/Recall".
2. In the "Save" tab, press the "Saveset" button .
3. Tap "Save" to save the settings to the specified file.
   Tap "Save As" to save the settings to a different file. Select the file and directory from the file selection dialog box.

   The current settings are saved to the selected file.

To load settings from a saveset file

Alternatively, you can use the "Recall" function on the toolbar.

1. Tap "Menu" and select "Save/Recall".
2. Select the "Recall" tab.
3. Press the "Saveset" button.
4. Tap "Load" to load the settings from the specified file.
   Tap "Open" to navigate to a different file. Select the file from the file selection dialog box and tap "Select".

   The saved settings are loaded to the R&S RTO6.

To save a One File

1. If you want to save reference waveforms in the One File, create and display the reference waveforms.
2. Tap "Menu" and select "Save/Recall".
3. In the "Save" tab, tap the "One File".
4. Tap "Save" to save the settings to the specified file.
   Tap "Save As" to save the settings to a different file. Select the file and directory from the file selection dialog box.

   The current settings and the active reference waveforms are saved to the selected file.

To load a One File

1. Tap "Menu" and select "Save/Recall".
2. Press the "One File" button.
3. Tap "Load" to load the settings from the specified file.

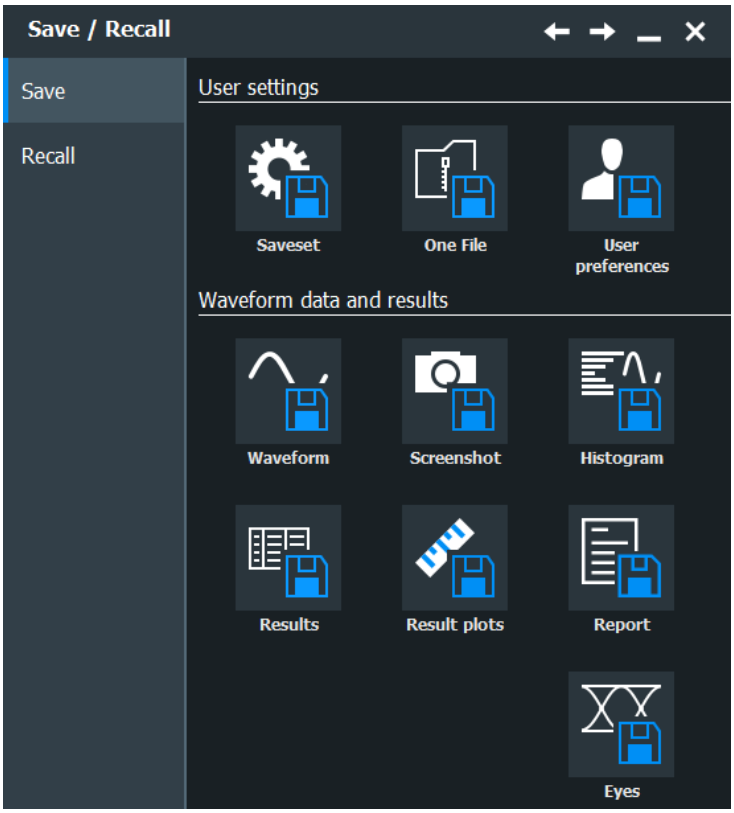
Tap "Open" to navigate to a different file. Select the file from the file selection dialog box and tap "Select".

The saved settings and reference waveforms are loaded to the R&S RTO6.

12.2 Save and recall waveform data and results

Access: "Menu" key >"Save/Recall" key > "Save" tab

You can export various data to file: waveform data, histograms, and measurement results.



For information on data export in I/Q mode (requires option), see Chapter 17.3, "I/Q data output", on page 1064.

- Waveform export files 453
- Waveform settings 460
- Histogram 465
- Results 467
- Result plots analysis 469
- Eyes 471
- Saving and loading waveform data 473

12.2.1 Waveform export files

Waveforms can be stored in XML, CSV, or BIN format.

Reloading waveforms: Restrictions

To reload waveform data as a reference waveform, it must be stored in BIN format.

If multiple acquisitions of one waveform are exported (Data logging or Multiple waveforms), only the first acquisition can be reloaded.

If raw data was exported, after reload the waveform position is not the same as for the exported waveform.

If the signal is a spectrum, reloading is only possible for waveforms with "Magnitude unit" = Linear. Waveforms with logarithmic unit cannot be reloaded.

Data of all waveforms is saved in two files. One file contains the waveform data values and is indicated by `*Wfm.*` in the filename. The second file contains the header data, for example, time scale, vertical scale, vertical and horizontal positions, interpolation mode and much more. Header data is required to restore the waveform from data, or to analyze the data values of the data file.

12.2.1.1 Header files

The header files of XML and BIN waveform files are written in XML format. The header files of CSV waveform files are written in CSV format. You can open the header files and use their information for data analysis.

CSV header files only contain the property names and values, one property per row.

```
Resolution:1e-010:
RecordLength:1000:
```

XML header files contain more information than CSV header files. The additional information is required to reload the stored waveforms with their correct settings.

```
<Prop Avail="0" ValueKey="" Name="Resolution" Value="1e-010" UserValue="0"
Step="1e-011" Default="0" Min="0" Max="1e+026" StepDefault="1e-011"
StepFactor="10" Resolution="0" UnitId="55" UnitName="s" UnitPowerProduct=""
BitGroupSize="0" Format="0"></Prop>

<Prop Avail="0" ValueKey="" Name="RecordLength" Value="1000" UserValue="1000"
Step="1" Default="1000" Min="0" Max="4294967295" StepDefault="1" StepFactor="10"
Resolution="1" UnitId="93" UnitName="Sa" UnitPowerProduct="" BitGroupSize="0"
Format="0"></Prop>
```

Header files contain the following properties:

Table 12-1: Header file properties

Value	Description
General	
FirmwareVersion	Firmware version that is installed on the R&S RTO6 (last entry in the header file)
Source	Name of the exported waveform
Resolution	Time between two samples *Resolution = 1 / Sample Rate*
SignalResolution	Time between two samples in this waveform. The value can differ from Resolution if the source is, for example, a spectrum, a bus signal, a correlation or a measurement. The value is determined automatically considering the waveform parameters and their dependencies. If the signal is a spectrum, the value indicates the frequency range of FFT bins.
EnhancementMode	Method to increase the sample rate if the required sample rate is higher than the ADC sample rate.
InterpolationMode	Interpolation method. The value is relevant when the enhancement mode is interpolated time.
DecimationMode	Method to reduce the number of data samples to achieve the required sample rate
DecimationFactor	Factor to the number of data samples to achieve the required sample rate *Decimation factor = ADC sample rate / Sample rate*
TraceArithmetics	Off, Envelope, or Average
InterleavedTraceCount	Number of y-values saved at each sampling time. The value is usually 1. The value is 2, if min and max values are saved for each sample, for example, for envelope waveforms.
SignalFormat	Format of the data values: • FLOAT: floating point numbers, general export format • INT (8 Bit): Integer 8 bit, used for "Raw (ADC sample)" data export. • INT (16 Bit): Integer 16 bit, used for "Raw (ADC sample)" data export in high definition mode.
Timestamp	Absolute time of the waveform recording
ByteOrder	Endianness, only relevant for raw data export in high definition mode (SignalFormat = INT (16 Bit)). • LSB first: little endian, least significant byte first • MSB first: big endian, most significant byte first
NumericFormat	Number format of bus values and digital channel data (bit pattern format)
Record length	
RecordLength	Number of samples in a waveform record of one acquisition
HWRecordLength	Equivalent to the RecordLength
SignalRecordLength	Number of required samples in the waveform. The value can differ from RecordLength and HWRecordLength if the source is, for example, a spectrum, a bus signal, a correlation or a measurement. The value is determined automatically considering the waveform parameters and their dependencies. If the signal is a spectrum, the value indicates the number of FFT bins.

Value	Description
SignalHardwareRecordLength	Number of samples actually available in this waveform, including the number of required samples in the waveform and the additional samples needed for further computation
LeadingSettlingSamples	Relevant only for BIN files. In XML and CSV files, the value is 0. Number of additional samples before the beginning of waveform samples. These additional samples are needed for further computation, for example, for filters.
Horizontal system	
TimeScale	Horizontal scale in seconds per division
HorizontalDivisionCount	Number of horizontal divisions
RescaleCenterTime	Horizontal position, the time distance between the reference point and the zero point of the diagram
RescaleCenterPoint	Position of the reference point in % of the screen
ReferencePoint	Position of the zero point in % of the screen
TriggerOffset	Time distance from the trigger point to the zero point of the diagram
XStart	Horizontal start value of the waveform (time or frequency) *)
XStop	Horizontal stop value of the waveform (time or frequency)
HardwareXStart	Actual horizontal start value of data, including the settling time for further computation *)
HardwareXStop	Actual horizontal stop value of data, including the settling time for further computation
	*) If the waveform is a spectrum, the XStart and HardwareXStart values may be slightly smaller than the specified start frequency, or even get negative. The spectrum is centered on the center frequency, and the frequency range covered by one spectral bin is given by the SignalResolution. Hence, the spectral bin in the center of the spectrum always covers the range [CenterFrequency; CenterFrequency + SignalResolution[. As a result, the range covered by the first spectral bin in the spectrum may reach further than the specified start frequency. It is ensured that the specified start frequency is included in the frequency range.
Vertical system If multiple channels are exported, the values of channel1 are delivered, no matter if channel 1 is exported or not.	
VerticalScale	Vertical scale of the waveform in Volts per division, or other unit / division
VerticalDivisionCount	Number of vertical divisions
VerticalPosition	Vertical position of the waveform in divisions
VerticalOffset	Vertical offset of the waveform in Volts, or other unit
NofQuantisationLevels	Theoretical number of quantization levels in the signal. This value depends on the waveform format (8 bit, 16 bit, …). In case of a math waveform, it depends on the quantization levels of the operands and on the operator type.
BaseYStart	Vertical start value of the waveform

Value	Description
BaseYStop	Vertical stop value of the waveform
Multi channel export The header files contain strings like this: `MultiChannelVerticalOffset: 4:1.63:1.96:0:0:1e-005:0:-1e+026:1e+026:1e-005:10:0:V:`. Only the first 5 values and the unit at the end of the string are relevant for data analysis. All other values are for internal use and not explained here. Examples are in csv format.	
MultiChannelExport	Indication whether multiple channels are exported simultaneously: On | Off
MultiChannelExportState	Number of channels and export status of the individual channels, for example, 4:On:Off:On:On...: channels 1, 3 and 4 are exported.
MultiChannelVerticalOffset	Number of channels and vertical offset of the individual channels, for example, 4:0:0:0.02:0...: channel 3 has an offset of 20 mV.
MultiChannelVerticalPosition	Number of channels and vertical position of the individual channels, for example, 4:0:0:0:2...: the position of channel 4 is 2 divisions.
MultiChannelVerticalScale	Number of channels and vertical scale of the individual channels, for example, 4:0.05:0:0.03:0.04...: scale of channel is 50 mV/div, channel 3 has 30 mV/div and channel 4 has 40 mV/div.
MultiChannelBaseYStart	Number of channels and minimum value of the vertical range for each individual channel, for example, 4:-0.25:0:-0.13:-0.28...
MultiChannelBaseYStop	Number of channels and maximum value of the vertical range for each individual channel, for example, 4:0.25:0:0.17:0.12...: The range of channel 1 is -250 mV to 250 mV. The range of channel 3 is -130 mV to 170 mV. The range of channel 4 is -280 mV to 120 mV.
History	
TimestampState	State of the timestamps export. If on, the timestamps of each history waveform are written to the waveform data file.
Math waveform	
BaseUnit	Base unit of a mathematic waveform, for example, linear unit
ViewUnit	User-selected unit of a mathematic waveform, for example, logarithmic unit for a spectrum. The value is only valid if the exported waveform is a math waveform.
ViewUnitRelative	Indication of a relative unit. It is true if the math waveform has the ViewUnit "dB", for example. The value is only valid if the exported waveform is a math waveform.
ViewReferenceLevel	Reference level for a relative unit. The value is only valid if the exported waveform is a math waveform, and the unit is relative.
FFT	
CenterFreq	Center frequency of the spectrum
FreqSpan	Frequency span of the spectrum
FrequencyStart	Start frequency of the spectrum
FrequencyStop	Stop frequency of the spectrum

Value	Description
WindowType	Window used for the spectrum computation
ResolutionBW	Resolution bandwidth of the spectrum
AdjustedResolutionBW	Actual resolution bandwidth of a spectrum waveform. The value is only valid if the exported waveform is a spectrum.
GateRBWCoupling	Indication whether the record length or the resolution bandwidth is a constant for the spectrum computation
Parameters for power calculation	
Impedance	Impedance used for power calculation
NoiseBandwidth	Noise bandwidth of a spectrum waveform, required for power calculation. The value is only valid if the exported waveform is a spectrum.
Parameters for internal use	
SourceType	Source qualifier
TraceType	Waveform qualifier
ValueType	
TOADone	
BaseUnitRelative	Base unit indication
UseInterSampleTriggerOffset	
ISO_TRG SC_POST SC_TRG	

12.2.1.2 Waveform data files

The waveform data files - indicated by `*Wfm.*` in the filename - contain the actual waveform data. Usually only Y-values - mostly voltage values - are written subsequently. If the signal is a spectrum, the data of the last frame is written.

If the waveform consists of minimum and maximum values, two Y-values per sample are written, and the property `InterleavedTraceCount` in the header file is >1. This applies to envelope waveforms, for example.

The option "Interleaved X/Y" allows you to include horizontal values into the file.

If multi-channel export is enabled, the Y-values of the selected channels are written in interleaved order.

- One channel, single acquisition export
  - Normal waveform:
    Y_0; Y_1; Y_2; Y_3; ...
  - Envelope waveform:
    $Ymin_0$; $Ymax_0$; $Ymin_1$; $Ymax_1$; $Ymin_2$; $Ymax_2$; $Ymin_3$; $Ymax_3$; ...
  - Normal waveform, interleaved x/y data:

X_0; Y_0; X_1; Y_1; X_2; Y_2; X_3; Y_3; ...

– Envelope waveform, interleaved x/y data:
X_0;$Ymin_0$; $Ymax_0$; X_1; $Ymin_1$; $Ymax_1$; X_2; $Ymin_2$; $Ymax_2$; X_3; $Ymin_3$; $Ymax_3$; ...

- Multi-channel, single acquisition export
In the example, two channels are exported.
  – Normal waveforms:
  $YCh1_0$; $YCh2_0$; $YCh1_1$; $YCh2_1$; $YCh1_2$; $YCh2_2$; $YCh1_3$; $YCh2_3$; ...
  – Envelope waveforms, channel 1 and channel 2 are envelopes:
  $YCh1min_0$; $YCh1max_0$; $YCh2min_0$; $YCh2max_0$; $Ymin_1$; $Ymax_1$; $YCh2min_1$; $YCh2max_1$; $Ymin_2$; $Ymax_2$; $YCh2min_2$; $YCh2max_2$; $Ymin_3$; $Ymax_3$; $YCh2min_3$; $YCh2max_3$; ...
  – Normal waveforms, interleaved x/y data:
  X_0; $YCh1_0$; $YCh2_0$; X_1; $YCh1_1$; $YCh2_1$; X_2; $YCh1_2$; $YCh2_2$; X_3; $YCh1_3$; $YCh2_3$; ...
  – Envelope waveform and normal waveform, interleaved x/y data:
  X_0;$YCh1min_0$; $YCh1max_0$; $YCh2_0$; X_1; $YCh1min_1$; $YCh1max_1$; $YCh2_1$; X_2; $YCh1min_2$; $YCh1max_2$; $YCh2_2$; X_3; $YCh1min_3$; $YCh1max_3$; $YCh2_3$; ...

In XML and CSV waveform value files, the data of each sample is grouped. The example shows the values of two samples for two waveforms and interleaved x/y data. The first waveform is an envelope, the second one is a normal waveform.

In CSV files, the data values for a given sampling time are written in one row.

```
-1.96e-008    -0.0079051387     -0.0059288535     -0.1027668
-1.95e-008    -0.0098814229     -0.0079051387     -0.10474309
```

In XML format, an empty line marks the beginning of the next sample.

```
<Data>-1.96e-008</Data>
<Data>-0.0079051387 </Data>
<Data>-0.0059288535 </Data>
<Data>-0.1027668 </Data>

<Data>-1.95e-008</Data>
<Data>-0.0098814229 </Data>
<Data>-0.0079051387 </Data>
<Data>-0.1027668 </Data>
```

If multiple acquisitions (Data logging / Multiple waveforms) are exported, the first acquisition is written in the same way as with single acquisition export. The following acquisitions are appended in the same way. If the signal is a spectrum, the last frame of each acquisition is saved.

If the history is exported in one file with timestamps, the timestamp as double float precedes the data of each acquisition. The acquisition data ist written in the same way as with single acquisition, according to the selected settings.

```
<timestamp as double float of first acquisition>
<value 1>
<value 2>
<value ...>
```

```
<timestamp as double float of second acquisition>
<value 1>
<value 2>
<value ...>
```

In BIN files, the instrument writes some leading and trailing settling samples before and after the waveform data. They ensure that the analysis of the reloaded reference waveform returns the same results as analysis of the original waveform. The number of leading settling samples is provided in the header file.

12.2.1.3 Number of samples in the waveform data file

In this section, a sample is defined as one or more values acquired at a given sampling time. The number of samples for one channel and acquisition is given in the header file by the property `SignalHardwareRecordLength`.

If the waveform has more than one Y-value per sample (e.g. envelope), the property `InterleavedTraceCount` is > 1, and the number of values in the file for this waveform is:

*No of values per waveform = InterleavedTraceCount * SignalHardwareRecordLength*

If multiple acquisitions are exported, the total number of values in the file is:

*No of values = InterleavedTraceCount * SignalHardwareRecordLength * No of exported acquisitions*

If "Interleaved x/y"is enabled, one horizontal value is added per sample. The total number of values in the file is:

*No of values = (1+ InterleavedTraceCount) * SignalHardwareRecordLength * No of exported acquisitions*

In BIN files, the value `SignalHardwareRecordLength` includes the number of required samples in the waveform and additional samples at the beginning (leading samples) and the end of the file (trailing samples). The number of additional samples is:

No of additional samples = SignalHardwareRecordLength - SignalRecordLength

The number of leading additional samples is given in the header file: `LeadingSettlingSamples`.

The number of trailing additional samples is:

No of trailing additional samples = No of additional samples - LeadingSettlingSamples
= SignalHardwareRecordLength - SignalRecordLength - LeadingSettlingSamples

MSO option R&S RTO6-B1:

If the data of digital channels is stored in BIN format, 1 bit is written for each sample. 8 data samples are written in 1 byte (data word). Thus, the file size is

File size = Number of samples / 8

For example, 100 MSa are written into a 12.5 MByte BIN file. After reading the file, you have to extract the samples from the data words.

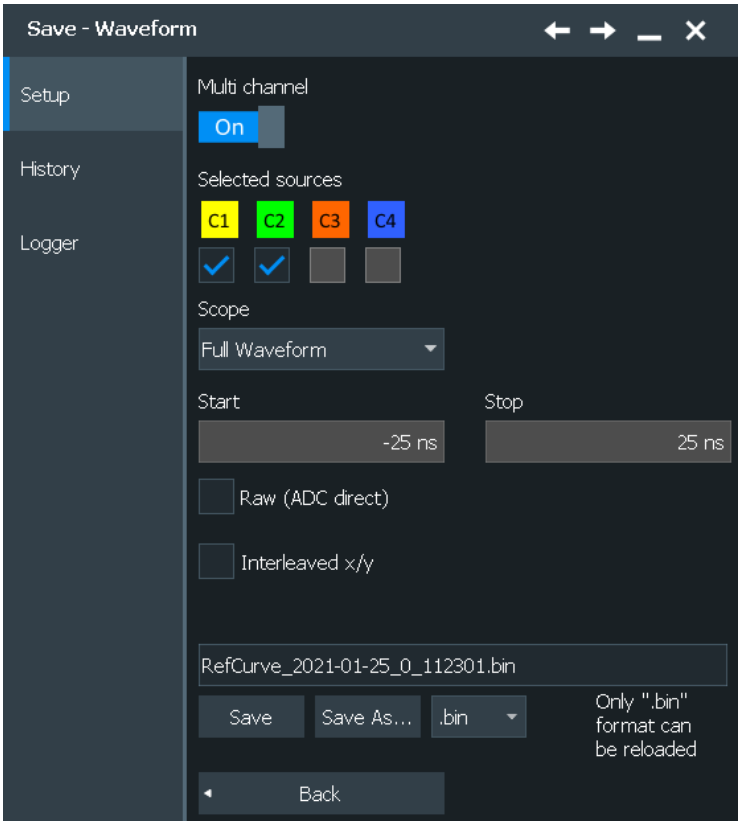
12.2.2 Waveform settings

In the "Save - Waveform" dialog, the storage settings for waveform data are defined.

See also: Chapter 12.2.7, "Saving and loading waveform data", on page 473.

12.2.2.1 Waveform setup settings

Access: "Menu" key >"Save/Recall" key > "Save" tab > "Waveform" > "Setup" tab.



Multi channel
Enables or disables the export of multiple input channels. If enabled, you can export the data of selected input channels (Selected sources) into one file.

If disabled, you can export one "Source" on page 460 waveform.

You can reload exported multiple channels if they are stored in BIN format. The import asks you to assign each stored waveform to a reference waveform.

Remote command:
`EXPort:WAVeform:MULTichannel` on page 1650

Source
Selects the waveform to be exported if "Multichannel export" is disabled.

Active waveforms of input channels, math signals, reference waveforms and advanced jitter analysis are available for export.

If the MSO option R&S RTO6-B1 is installed, you can save also digital channels and parallel buses.

Remote command:
`EXPort:WAVeform:SOURce` on page 1649

Selected sources
Select the channels to be included in data export if "Multi channel" is enabled. Waveform1 of up to four input channels can be saved into one file.

Remote command:
`CHANnel<m>:EXPortstate` on page 1651

Scope
Defines the part of the waveform record that has to be stored.

"Full waveform"	Saves the complete waveform record.
"Zoom"	Saves the data included in the zoom area if at least one zoom is defined for the source waveform. The start and stop values of the area are shown. If several zooms are defined, select the "Zoom" to be used for export.
"Cursor"	Saves the data between the cursor lines if at least one cursor measurement is defined for the source waveform. The start and stop values of the area between the cursor lines are shown. If several cursor sets are defined, select the "Cursor set" to be used for export.
"Gate"	Saves the data included in the measurement gate if a gated measurement is defined for the source waveform. Select the "Measurement" for which the required gate is defined. The start and stop values of the gate are shown.
"Manual"	Saves the data between user-defined "Start" and "Stop" values.

Remote command:
`EXPort:WAVeform:SCOPe` on page 1652
`EXPort:WAVeform:STARt` on page 1652
`EXPort:WAVeform:STOP` on page 1652
`EXPort:WAVeform:ZOOM` on page 1653
`EXPort:WAVeform:CURSorset` on page 1653
`EXPort:WAVeform:MEAS` on page 1653

Raw (ADC direct)
Enables the export of analog channel data in the raw sample format of the ADC. The data format is integer 8 bit (signed 8-bit binary format). This format reduces the file size (1 Byte/sample instead of 4 Bytes/sample in binary files) but decreases the precision of the values.

If the high definition mode is active, the data format is integer 16 bit, except for peak detect decimation (8 bit). See "Export" on page 136.

Only y-values are exported, the "Interleaved x/y" option is not available.

Currently, the setting is not available for the export of digital channel data and data of R&S RT-ZVC channels.

Data conversion:

To convert INT8 or INT16 data to physical quantities, e.g. voltages, use the following formulas:

*ConversionFactor = VerticalScale * VerticalDivisionCount / NofQuantisationLevels*

*PhysicalQuantity = (Value_ADC * ConversionFactor) + VerticalOffset - (VerticalScale * VerticalPosition)*

The raw values are written in the `*.Wfm.*` file, all other values can be found in the corresponding header file.

Table 12-2: Example of raw data conversion

	INT8	INT16, HD mode
VerticalScale	110 mV/div	50 mV/div
VerticalPosition	1 div	0
VerticalOffset	50 mV	0
NofQuantisationLevels	253	253 * 256
VerticalDivisionCount	10	10
Value_ADC	13	-61
ConversionFactor	0.11 * 10 / 253 = 0.004347826086957	0.05 * 10 / (253 * 256) = 0.0000771986
Voltage	(13 * 0.004347826086957) + 0.05 - (1 * 0.11) = -0.003478260869559 V = 3.478260869559 mV	(-61 * 0.0000771986) + 0 = -4.7091146 mV

When reloading an exported raw BIN waveform, the waveform position after reload is not the same as for the exported waveform.

Remote command:
`EXPort:WAVeform:RAW` on page 1655

Interleaved x/y
Includes horizontal values in the export data (time or frequency values, depending on the waveform). X and Y-values are written alternately to the file. If disabled, only Y-values - mostly voltage values - are written.

Interleaved x/y data cannot be exported as raw values, the "Raw (ADC direct)" option is not available.

Remote command:
`EXPort:WAVeform:INCXvalues` on page 1655

Save to file
Enter the filename to save the waveform to. Double-tap the filename to open the file selection dialog box.

By default, the filename has the prefix "RefCurves_". You can define a pattern for automatic naming in the "Autonaming" dialog, see Chapter 4.8.1, "Autonaming", on page 104.

"Save" Saves the waveform as a reference waveform in the selected file.

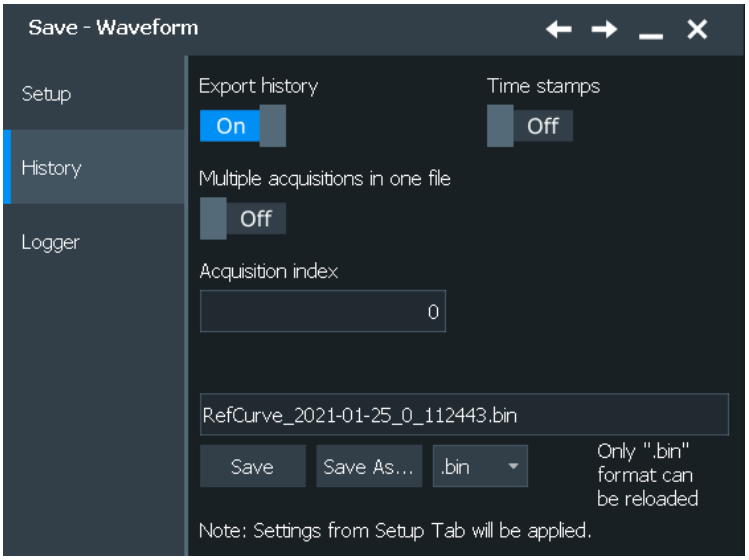
"Save As..." Opens the file selection dialog box and saves the waveform to the selected file. See also Chapter 12.5, "File selection dialog", on page 482

".bin/.xml/.csv" Selects the file format. Note that reference waveforms can be loaded from `.bin` files only.
See also: Chapter 12.2.1, "Waveform export files", on page 453.

Remote command:
`EXPort:WAVeform:NAME` on page 1651
`EXPort:WAVeform:SAVE` on page 1651

12.2.2.2 Waveform history settings

Access: "Menu" key >"Save/Recall" key > "Save" tab > "Waveform" > "History" tab.



In this dialog you can configure the export of the history waveform data to file. Select the channels and scope in the "Setup" tab.

Export history
Enables the history mode and the export of history waveforms to file.

Remote command:
`CHANnel<m>[:WAVeform<n>]:HISTory[:STATe]` on page 1461

Time stamps
If enabled, the relative timestamps of all history waveforms are written into the waveform data file at the beginning of each waveform record.

The following format is used:

```
<timestamp as double float of first acquisition>
<value 1>
<value 2>
<...>
<timestamp as double float of second acquisition>
```

```
<value 1>
<value 2>
<...>
```

You can also select the format of the time stamp in the "Acquire" > "History" dialog.

Remote command:
`EXPort:WAVeform:TIMestamps` on page 1654

Multiple acquisitions in one file
If enabled, allows you to save several or all history waveforms. Define the part of the history to be exported using "Start acquisition" and "Stop acquisition".

If multiple acquisitions of one waveform are exported into a BIN file, the first acquisition can be reloaded as reference waveform.

Remote command:
`CHANnel<m>[:WAVeform<n>]:HISTory:STARt` on page 1462
`CHANnel<m>[:WAVeform<n>]:HISTory:STOP` on page 1462

Start export
Starts the history replay and simultaneous saving.

Remote command:
`CHANnel<m>[:WAVeform<n>]:HISTory:PLAY` on page 1463

Save to file
Enter the filename to save the waveform to. Double-tap the filename to open the file selection dialog box.

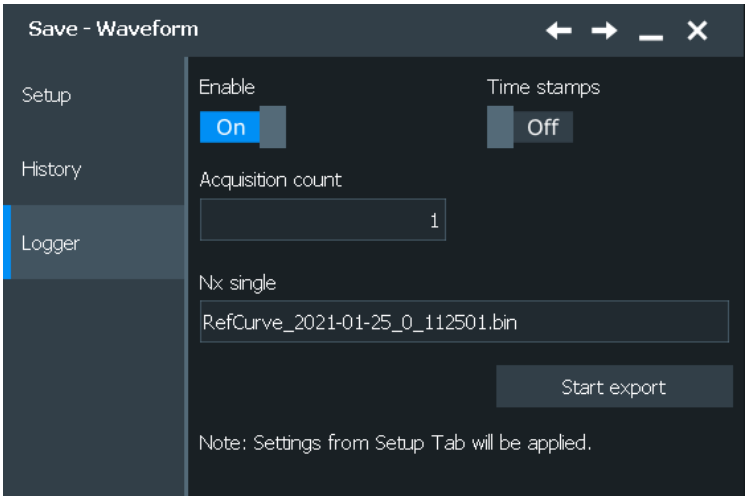
By default, the filename has the prefix "RefCurves_". You can define a pattern for automatic naming in the "Autonaming" dialog, see Chapter 4.8.1, "Autonaming", on page 104.

"Save"	Saves the waveform as a reference waveform in the selected file.
"Save As..."	Opens the file selection dialog box and saves the waveform to the selected file. See also Chapter 12.5, "File selection dialog", on page 482
".bin/.xml/.csv"	Selects the file format. Note that reference waveforms can be loaded from `.bin` files only. See also: Chapter 12.2.1, "Waveform export files", on page 453.

Remote command:
`EXPort:WAVeform:NAME` on page 1651
`EXPort:WAVeform:SAVE` on page 1651

12.2.2.3 Waveform logger settings

Access: "Menu" key >"Save/Recall" key > "Save" tab > "Waveform" > "Logger" tab.



In this dialog, you can configure the export of the data of a running Nx Single acquisition. Select the channels and scope in the "Setup" tab.

Enable
Enables the export of all waveforms of an Nx Single acquisition into one file.

The waveform records are written in historical order one after the other, either the complete records or the sections as defined in Scope. Set the number of acquisitions to be acquired and stored with "Acquisition count". The maximum amount of data that can be written is shown in "Max. file size".

Enabling "Data logging" stops a running acquisition. To start the logging, tap "Start Export" on page 465 or press [Single].

Pressing "Run cont" disables data logging.

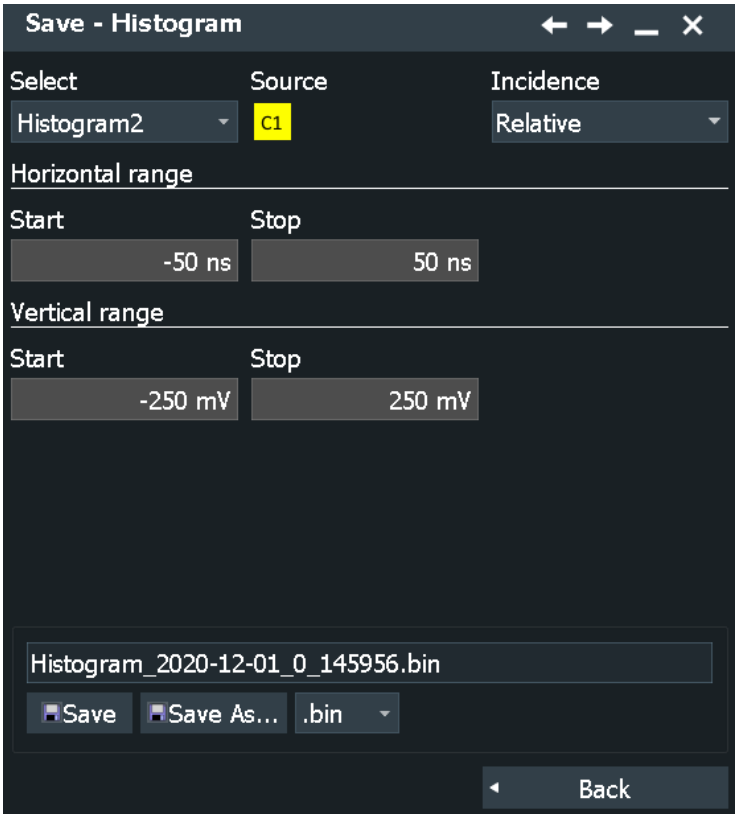
Remote command:
`EXPort:WAVeform:DLOGging` on page 1654

Start Export
Starts an Nx Single acquisition series and simultaneously saves the waveform data to a file.

Remote command:
`RUNSingle` on page 1325 (Nx Single acquisition)

12.2.3 Histogram

Access: "Menu" key >"Save/Recall" key > "Save" tab > "Histogram"



The waveform histogram export saves data in two files. The `*.Wfm.*` file contains 256 or 512 absolute or relative histogram values. The other file is the header file.

Contents of the header file:

- Source waveform of the histogram
- Histogram mode: vertical or horizontal
- Incidence of exported values: absolute or relative
- Histogram range: XStart, XStop, YStart, YStop
- Name of the exported histogram

Data conversion:

Using the header data, you can calculate the waveform value to which a histogram value belongs:

*YValue = (YStop - YStart) / HistogramValuesCount * HistogramValueNumber + YStart*

YStart	-0.25 V
YStop	0.25 V
HistogramValuesCount	256 (total number of written rows in a CSV file)
HistogramValueNumber	68 (number of the row in a CSV file)
Y-Value	(0.25 - (-0.25)) / 256 * 68 - 0.25 = -0.11719 V

The histograms settings are defined in the "Histogram" dialog box ("Menu" > "Apps" > "Analysis" tab > "Histogram").

See also: Chapter 8.2.8.4, "Histogram setup", on page 345.

Select
Selects the histogram to be exported. All active waveform histograms are shown in the list.

Measurement histograms can also be exported, see Chapter 12.2.5, "Result plots analysis", on page 469.

Remote command:
`EXPort:HISTogram:SELect` on page 1656

Source
Displays the source of the selected histogram.

Incidence
Sets the mode of exported histogram data: relative or absolute count of values. If relative values are exported, the sum of all values is 1, and the count of each value is set in relation to the sum.

Remote command:
`EXPort:HISTogram:INCidence` on page 1656

Horizontal range, Vertical range
Displays the "Start" and "Stop" limits of the histogram area.

Save to file
Enter the filename to save the waveform histogram to. Double-tap the filename to open the file selection dialog box.

By default, the filename has the prefix "Histogram_". You can define a pattern for automatic naming in the "Autonaming" tab. The default directory is:

`C:\Users\Public\Documents\Rohde-Schwarz\RTO6\Histograms`

"Save"	Saves the histogram data in the selected file.
"Save As..."	Opens the file selection dialog box and saves the histogram data to the selected file. See also Chapter 12.5, "File selection dialog", on page 482
".bin/.xml/.csv"	Selects the file format.

Remote command:
`EXPort:HISTogram:NAME` on page 1657
`EXPort:HISTogram:SAVE` on page 1657
`EXPort:HISTogram:DATA?` on page 1657

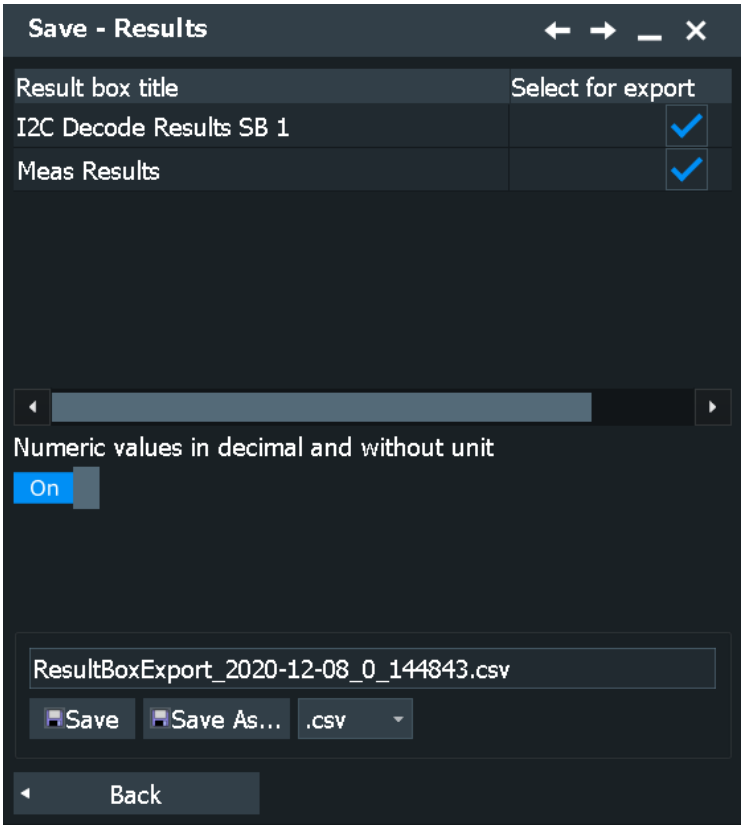
12.2.4 Results

In this dialog, you can select the result boxes to be saved, and define the storage settings.

Access: "Menu" key >"Save/Recall" key > "Save" tab > "Results"

Access to the tab is available in all tabs where measurement and analysis settings are defined, for example, in the "Measurements Setup", "Cursors Setup", and "Masks Test Definition" tabs: Simply tap the "Export" button.

Furthermore, you can export the decode result tables of serial protocol analysis.



Result selection
The table lists all result boxes and decode tables that are currently open, including minimized boxes and docked boxes. Select the results that you want to save to file. All results are written into one file.

Note: If the result box is minimized, only the columns shown on the result icon are saved (2 columns). Statistical results are not shown on the minimized results icon, and they are not saved.

Numeric values in decimal and without unit
By default, numeric result values are written with their unit to the file. If the option is enabled, the values are saved with more decimal places.

Export results
Enter the filename to save the results to. Double-tap the filename to open the file selection dialog box.

By default, the filename has the prefix "ResultBoxExport_".

You can define a pattern for automatic naming in the "Menu" > "Settings" > "Save/Recall" > "Autonaming".

"Save" Saves the selected results to the indicated file.

"Save As..." Opens the file selection dialog box and saves the selected results to the selected file. See also Chapter 12.5, "File selection dialog", on page 482

".csv/.html" Selects the file format.

- CSV: the values are saved in a file
  You can select the value delimiter and the list separator symbol in the "CSV Export" dialog, see Chapter 4.8.2, "CSV export", on page 107.
  Tip for using MS Excel: It is recommended that you use the semicolon as csv decimal symbol. When you open the file with MS Excel, use "File > Open" and follow the wizard to set the separators correctly, or set the separator settings with "Tools > Options > International".
- HTML: Results are saved as web page for display in a browser.

12.2.5 Result plots analysis

Access: "Menu" key >"Save/Recall" key > "Save" tab > "Result Plots"

You can export the data of long-term measurements, the measurement histogram and track data to file.

The measurement export saves results in two files. The `*.Wfm.*` file contains data values, and the other file is the header file.

The header file contains:

- Source waveform of the measurement
- Measurement scale
- Export type = Histogram, Long term or Track
- Exported measurement
- Histogram range: XStart, XStop, YStart, YStop
  The range is only relevant for export type = histogram. The measurement axis is the X-axis, which can be a horizontal or vertical axis depending on the histogram mode.

Long-term measurements: The `*.Wfm.*` file contains one value or value set for each long-term measurement point. The maximum number of points is defined in the "Horizontal scaling" dialog box.

- If statistics are disabled, the current result of the main measurement is written - one double value per long-term point.
- If statistics are enabled, seven values for each long-term point are saved:
  - Upper peak
  - Lower peak
  - RMS

- Standard deviation
- Average
- Event count per point: number of measurement results that creates one long-term point
- Waveform count per point: number of waveforms included in one long-term point.

Measurement histogram: The `*.Wfm.*` file contains 1000 absolute or relative histogram values.

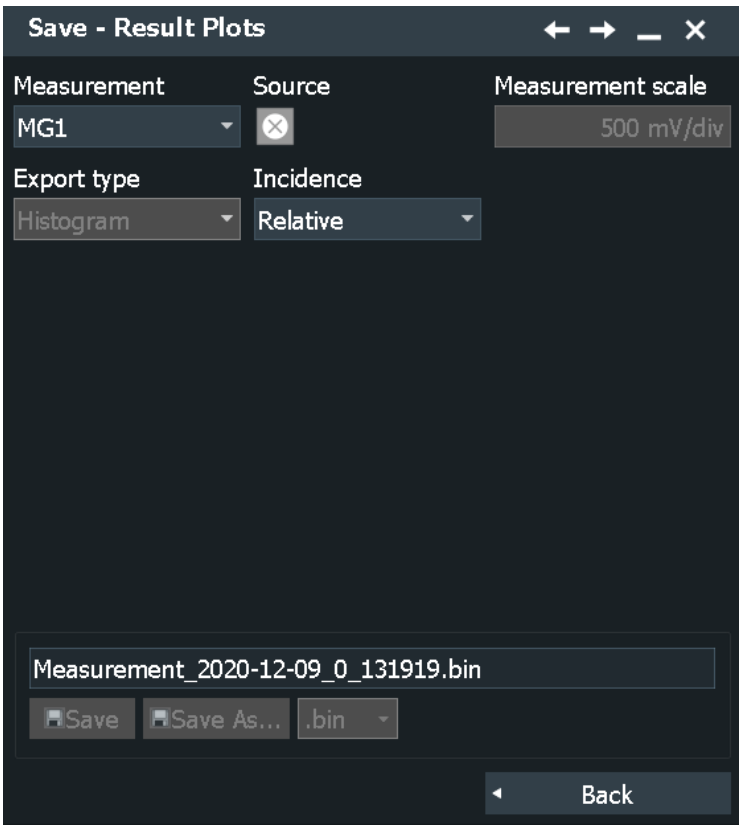
Data conversion of measurement histogram data:

Using the header data, you can calculate the measurement value to which a histogram value belongs:

*MeasValue = (XStop - XStart) / 1000 * HistogramValueNumber + XStart*

Example: The 273rd histogram value is 0.491749. That means, the relative frequency of the measurement value 0.1246 V is 0.491749.

XStart	0.07 V
XStop	0.27 V
HistogramValueNumber	273 (number of the row in a CSV file)
MeasValue	(0.27 - 0.07) / 1000 * 273 + 0.07 = 0.1246 V



Measurement
Selects the measurement to be exported.

Remote command:
`EXPort:MEASurement:SELect` on page 1659

Source, Measurement scale
Displays the measurement settings source and scale.

Export type
You can export the result data of the long-term measurement, the measurement histogram or the track.

To export the data, the required type must be enabled in the ""Menu" > "Measure" > Plot" tab: "Long term > Enable", "Histogram > Enable" or "Track > Enable".

Remote command:
`EXPort:MEASurement:TYPE` on page 1659

Incidence
Sets the mode of exported histogram data: relative or absolute count of values. If relative values are exported, the sum of all values is 1, and the count of each value is set in relation to the sum.

Remote command:
`EXPort:HISTogram:INCidence` on page 1656

Save to file
Enter the filename to save the measurement data to. Double-tap the filename to open the file selection dialog box.

By default, the filename has the prefix "Measurement_".

You can define a pattern for automatic naming in the "Autonaming" tab, see Chapter 4.8.1, "Autonaming", on page 104.

The default directory is:

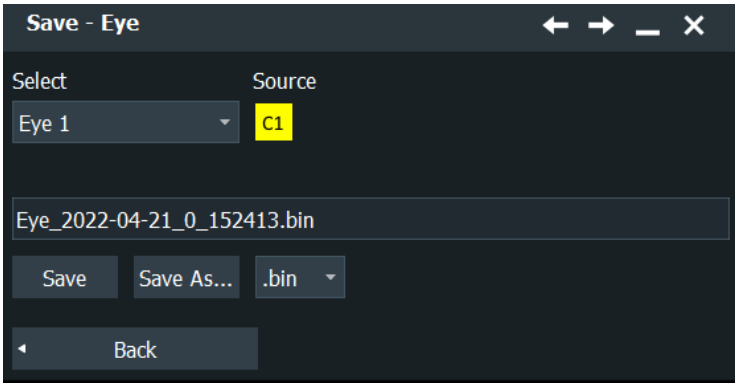
`C:\Users\Public\Documents\Rohde-Schwarz\RTO6\Measurements`

"Save"	Saves the measurement data in the selected file.
"Save As..."	Opens the file selection dialog box and saves the measurement data to the selected file. See also Chapter 12.5, "File selection dialog", on page 482
".bin/.xml/.csv"	Selects the file format.

Remote command:
`EXPort:HISTogram:NAME` on page 1657
`EXPort:MEASurement:SAVE` on page 1660
`EXPort:MEASurement:DATA?` on page 1660

12.2.6 Eyes

Access: "Menu" key >"Save/Recall" key > "Save" tab > "Eyes"



In this dialog, you can select an eye diagram that you want to export.

Eye export saves the accumulated values at each pixel of an eye diagram.

The export supports *.csv and *.bin files. Each export generates a header file and a data file.

Content and format of the *.csv files

The header file contains the following values:

- Name of the exported eye (e.g. Eye 1)
- Corresponding source (e.g. C1W1)
- Timestamp
- Number of lines and columns
- Vertical / horizontal range

The data file contains the comma separated list of the pixel values. The number of lines and columns corresponds to the number of lines and rows of the eye diagram image.

Content and format of the *.bin files

The header file contains the following values:

- Name of the exported eye (e.g. Eye 1)
- Corresponding source (e.g. C1W1)
- Timestamp
- Number of lines and columns
- Vertical / horizontal range

The data file contains the following information in the given order

- Number of bytes per value - uint32
- Number of samples – uint32
- Column wise sorted pixel values – ‘number of samples’ * ‘number of bytes per value’

Select/ Source
Selects the eye measuremen to be exported and displys its source.

Remote command:
`EXPort:EYE:SELect` on page 1658
`EXPort:EYE:SOURce?` on page 1658

Save to file
Enter the filename to save the eye data to. Double-tap the filename to open the file selection dialog box.

By default, the filename has the prefix "Eye_".

You can define a pattern for automatic naming in the "Autonaming" tab, see Chapter 4.8.1, "Autonaming", on page 104.

The default directory is:

`C:\Users\Public\Documents\Rohde-Schwarz\RTO6\Eyes`

"Save"	Saves the measurement data in the selected file.
"Save As..."	Opens the file selection dialog box and saves the measurement data to the selected file. See also Chapter 12.5, "File selection dialog", on page 482
".bin/.csv"	Selects the file format.

Remote command:
`EXPort:EYE:NAME` on page 1658
`EXPort:EYE:SAVE` on page 1658
`EXPort:EYE:DATA?` on page 1658

12.2.7 Saving and loading waveform data

You can save the data of a channel, math or reference waveform to an `.xml`, `.csv`, or `.bin` file. The data export of several channels into one file is also possible. Files in `.bin` format can be reloaded to the R&S RTO6 as reference waveforms.

Instead of a complete waveform, you can also save a part of it, limited by a previously defined zoom, cursor lines, measurement gate or user-defined time values.

To save waveform data quickly, you can add the "Save Waveform" icon to the toolbar and use it for saving. The icon does not work for saving actions that are started with "Start export" (data logging and multiple history waveforms).

It is also possible to save history data to file. Furthermore, you can save a "live record" of a running RUN Nx SINGLE acquisition to one data file.

For details on waveform save/recall settings, see Chapter 12.2.2, "Waveform settings", on page 460.

The following procedures are described:

- "To save a waveform or a part of a waveform to a file" on page 474
- "To save a waveform using the toolbar icon" on page 474
- "To export waveform data of a running acquisition" on page 474
- "To save the history data" on page 280
- "To load a reference waveform" on page 251

"To save a reference waveform" on page 250

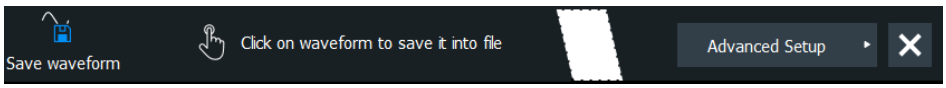
To save a waveform or a part of a waveform to a file

1. Open "Menu" and select "Save/Recall" > "Save".
2. Tap the "Waveform" button.

   The "Save - Waveform" dialog opens.
3. Select the waveforms to be saved:
   - To save one waveform, tap the "Source" icon and select the waveform.
   - To save data of several channels, enable "Multi channel" and select the channels.
4. In the "Scope" list, select the part of the waveform record to be saved. Zoom, cursor and gate segments require the same setup for the selected waveform before saving. For "Manual", enter the "Start" and "Stop" time of the section.
5. Check the filename under "Save to file" and change it if needed. Usually, autonaming is used.
6. Check the file format and the "Export format" settings and change them if needed.
7. Tap "Save" to save the waveform data to the specified file.
   Tap "Save As" to save the waveform data to a different file or file type. Select the file from the file selection dialog box.

To save a waveform using the toolbar icon

1. Add the "Save Waveform" icon to the toolbar, see Chapter 3.3.6.2, "Configuring the toolbar", on page 53.
2. Set the scope, export format, and other parameters in "Menu" > "Save/Recall" > "Waveforms".
3. If necessary, adjust the autonaming pattern and the storage path in "Menu" > "Settings" > "Save/Recall" > "Autonaming".
4. To save the waveform data:
   a) Tap the "Save Waveform" icon on the toolbar. The "Save Waveform" overlay menu opens.

   b) Tap the waveform you want to save.
   If you tap the diagram background, the data of the focused waveform is saved.

To export waveform data of a running acquisition

1. Select the waveform you want to save and the scope as described in "To save a waveform or a part of a waveform to a file" on page 474, step 1 to 5.
2. If you want to save only a section of each waveform, set the "Scope".

3. Open the "Logger" tab.
4. Tap "Enable".
5. Enter the number of acquisitions to be acquired and saved in "Acquisition count".
6. Check the filename under "Nx single" and change it, if needed. Usually, autonaming is used.
7. Tap "Start export" to start the acquisition and to save the acquired waveform data to the specified file.

To load waveform data as a reference waveform

To reload waveform data from a previous measurement, the waveform must have been stored as a reference waveform in a BIN file before.

The procedure is described in Chapter 7.2.1, "Working with reference waveforms", on page 249.

12.3 Screenshots

To store the graphical results of the measurement, you can save a screenshot of the graphic area. To document current settings, the open dialog box can be included in the screenshot.

The "Image" toolbar icon saves the current display to a file according to the settings in "Menu" > "Save/Recall" > "Save" tab > "Screenshot". See also "Image" on page 55.

You can configure the [Camera] key to save screenshots by a single keypress. See also Chapter 4.5.1, "Hardkeys: function assignment", on page 94.

If a USB flash drive is connected to the instrument, the default path of the user data directory is set to the drive letter of the USB flash drive. Thus, you save data to USB flash drive automatically, and you can change the directory in the file explorer at any time.

Screenshots on a computer using the Web interface

If the R&S RTO6 is connected to a LAN, you can create and save screenshots of the instrument's display on a computer. See Chapter 22.3.2, "Web browser", on page 1233.

Meta information in screenshots

The meta data of the screenshot also contains instrument information. In PNG and JPEG files, meta information is saved as EXIF information and can be read, for example, using the ExifTool.

Example:

Reading meta information using the ExifTool.

Command: `# exif C:\Screenshot_2020-07-14_0_110551.png`

Result:

```
ExifTool Version Number         : 10.20
File Name                       : Screenshot_2016-07-14_0_110551.png
Directory                       : C:/
File Size                       : 37 kB
File Modification Date/Time     : 2020:07:14 11:05:51+02:00
File Access Date/Time           : 2020:07:14 11:05:51+02:00
File Creation Date/Time         : 2020:07:14 11:05:51+02:00
...
Instrument Firmware Version     : 3.30.0.46
Instrument Material Number      : 1329.7002k44
Instrument Serial Number        : 123456
Image Size                      : 1280x800
Megapixels                      : 1.0
```

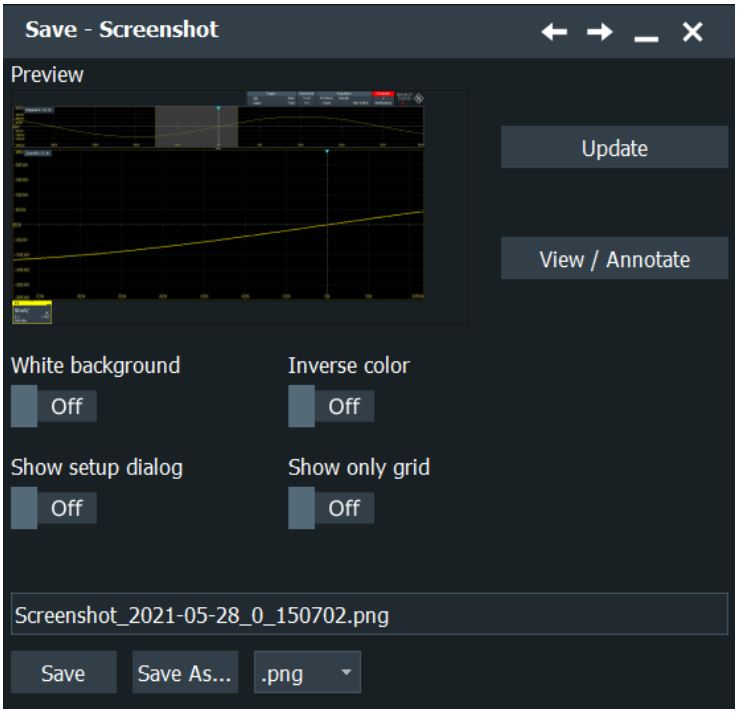
12.3.1 Screenshot settings

Access: "Menu" > "Save/Recall" > "Save" tab > "Screenshot"

In the "Screenshot" dialog box, you configure the image to be saved or included in a report. You select the storage location for screenshot files. The image is created when you open the dialog box, and can be updated at any time.

You can also edit the image before saving, and include an open dialog box or the sidebar in the image.

You can save the image in the "Screenshot" dialog box. To save screenshots quickly, use the "Image" toolbar icon, or configure and use the [Camera] key.



Preview 477
Update 477
White background 477
Inverse color 478
Show setup dialog 478
Show only grid 478
View/ Annotate 478
Save to file 478

Preview
Shows a preview of the screenshot. The image is created when the dialog box opens.

Update
Updates the preview of the screenshot with the current display view, e.g. after changes to the settings have been made, or an additional channel has been activated.

White background
Inverts the background color. So you can picture waveforms with normal waveform colors on white background.

If both "White background" and Inverse color are enabled, the instrument inverts the background twice, and it appears black.

"White background"	"Inverse color"	Background	Waveform and results
On	Off	White	Screen colors
Off	On	White	Inverted colors

"White background"	"Inverse color"	Background	Waveform and results
On	On	Black	Inverted colors
Off	Off	Black	Screen colors

Remote command:
`HCOPy:WBKG` on page 1662

Inverse color
Inverts the colors of the output, i.e. a dark waveform is shown on a white background.

See also: "White background" on page 477.

Remote command:
`HCOPy:DEVice<m>:INVerse` on page 1662

Show setup dialog
If you want to save dialog boxes in screenshots, enable "Show setup dialog". The currently open dialog box is included in the screenshot.

Remote command:
`HCOPy:SSD` on page 1663

Show only grid
If enabled, the screenshot shows only the grid. The signal bar is not included.

Remote command:
`HCOPy:ISBA` on page 1663

View/ Annotate
Opens the screenshot in the Paint application. Edit the image as necessary. You can store the file using "Save as". Alternatively, save the file and close the Paint application to return to the "Screenshot" dialog, then save the edited image. The changes are not shown in the preview.

Save to file
Defines the filename to which screenshot is saved. By default, the filename has the prefix "Screenshot_". Double-tap the filename filed to change the name.

If a USB flash drive is connected to the instrument, the default path of the user data directory is set to the drive letter of the USB flash drive. Thus, you save data to USB flash drive automatically, and you can change the directory in the file explorer at any time.

"Save" Saves the current screenshot to the specified file.

"Save As..." Opens the file selection dialog box. Here you can adjust the target directory and the file name and save the current screenshot to the file. The symbols of important target folders are listed on the left of the file explorer.

"Delete" Opens the file selection dialog box and deletes the selected file.

Remote command:
`HCOPy:DEVice<m>:LANGuage` on page 1662
`HCOPy:DESTination<1..2>` on page 1661
`MMEMory:NAME` on page 1661
`HCOPy:IMMediate<m>[:DUM]` on page 1664
`HCOPy:IMMediate<m>:NEXT` on page 1664
`MMEMory:DELete` on page 1645

12.3.2 Configuring and saving screenshots

You can edit the image, invert all colors, and set the background color. A preview of the current image is shown for reference.

1. Open the "Menu" > "Save/Recall" > "Save" tab > "Screenshot".
2. To enhance the images for later print on white paper, enable "White background" or "Inverse color". If you print this image later on a monochrome printer, you get a grayscaled picture. The contrast of the resulting gray lines depends on waveform colors and the used printer.
3. Select the file format: png, jpg, or another one.
4. To change the directory, tap "Save As" and configure the path.

   The symbols of often used target folders are listed on the left of the file explorer. By default, screenshots are saved in the `C:\Users\Public\Public Documents\Rohde-Schwarz\RTx\ScreenShots` directory.
5. Tap "Save".

   The file is saved and the dialog box closes.
6. Check if the screenshot is saved to the desired directory.
7. To save further screenshots, use one of the following ways:
   - Configure the [Camera] key. Press the key to save a screenshot.
     See also Chapter 4.5.1, "Hardkeys: function assignment", on page 94.
   - Add the "Image" icon to the toolbar. Tap the icon to save an image.
     See also Chapter 3.3.6.2, "Configuring the toolbar", on page 53.
   - Tap "Save" in the "Screenshot" dialog box to save the image to the specified file.
   - To save the image with a dedicated filename or to another directory, tap "Save As" in the "Screenshot" dialog box. Select the path, enter a filename, and tap "Save".

12.4 Reports

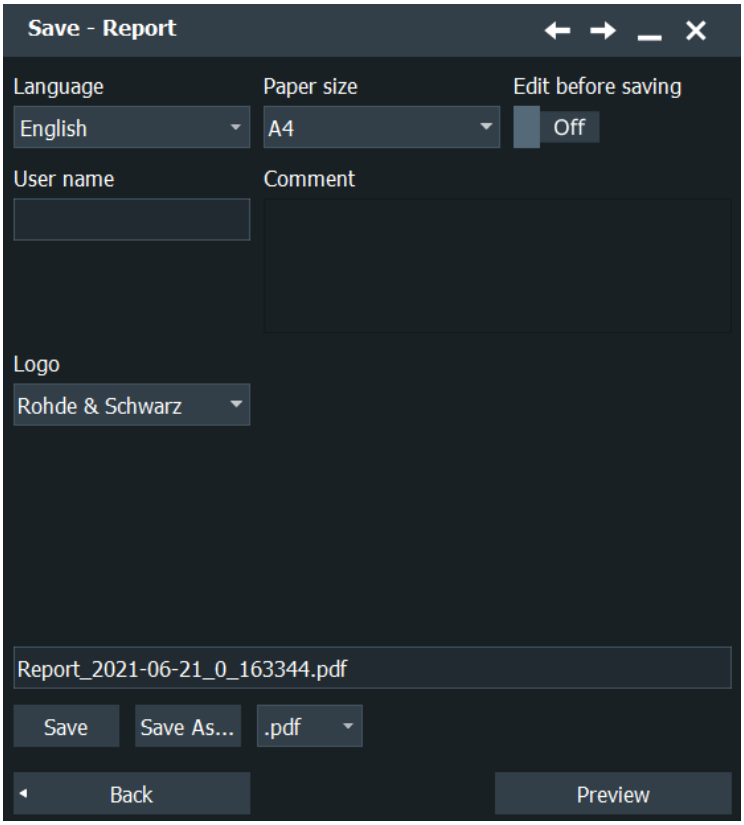
Reports document the current measurement and test results. The report contains general information, current vertical and horizontal settings, trigger settings, active channels and all current results except for zoom and search results. A screenshot is also included.

You can create the report manually, or automatically on defined events:

- Press the [Camera] key.
- Tap the "Create report" toolbar icon.
  Before, add the icon to the toolbar, see also Chapter 3.3.6.2, "Configuring the toolbar", on page 53.
- Action on micro button, available on active Rohde & Schwarz probes
- Action on trigger
- Event action at mask testing

12.4.1 Report settings

Access: "Menu" menu > "Save/Recall" menu > "Save" tab > "Report"



Language
Selects the language to be used in the report. Available languages are listed in the data sheet.

Remote command:
`REPort:LANGuage` on page 1665

Paper size
Selects the paper size: A4 or US Letter.

Remote command:
`REPort:PAPersize` on page 1665

Edit before saving
Enables you to edit the report info when you save reports using the [Camera] key. When you press the key, a dialog box opens where you can change the user name and the comment.

User name / Comment
Enter information that appears in the general information section at the beginning of the report.

Remote command:
`REPort:USER` on page 1666
`REPort:COMMent` on page 1666

Logo
By default, the Rohde & Schwarz logo is shown in the header of the report pages. You can switch the logo off, or select your logo to be shown. A preview of the selected log file is shown.

Remote command:
`REPort:LOGType` on page 1665
`REPort:LOGO` on page 1665

Save to file
Select the file format and define the filename of the report file. By default, the filename has the prefix "Report_".

Double-tap the filename to open the file selection dialog box.

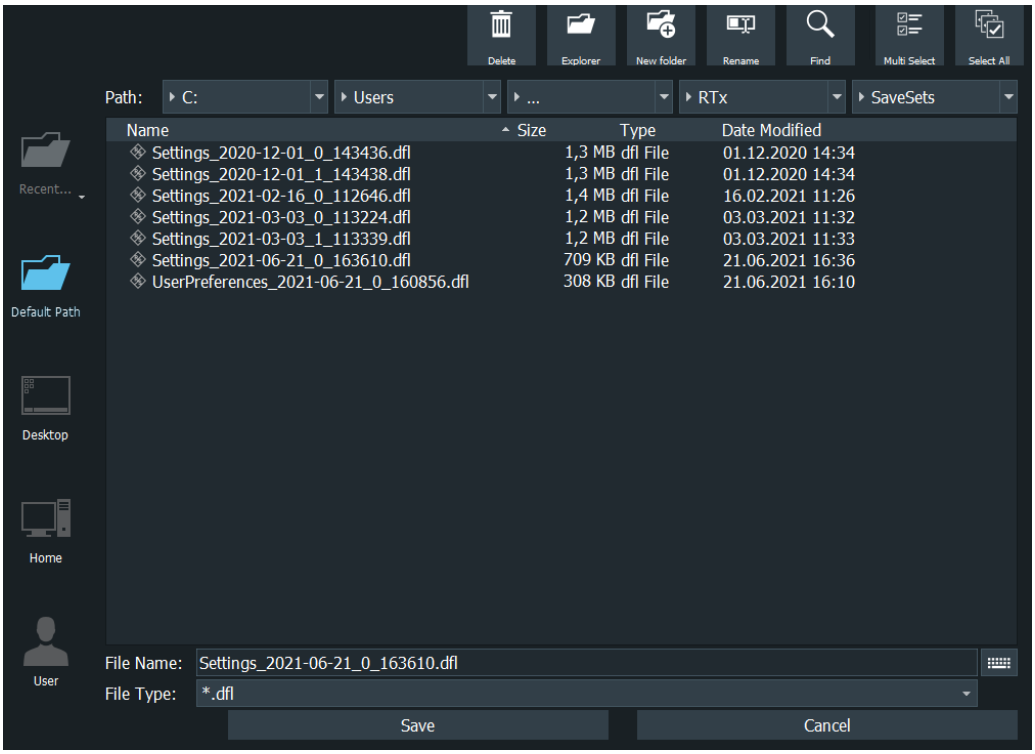
"pdf/doc/html"	Selects the report format.
"Save"	Saves the current report to the specified file.
"Save As..."	Opens the file selection dialog box and saves the report to the selected file.

Remote command:
`REPort:FILE:NAME` on page 1666
`REPort:FILE:SAVE` on page 1666

Preview
Opens the current report in PDF format.

12.5 File selection dialog

The file selection dialog provides a file explorer from which you can select a file to load or to save data to. You can also manage your files in this dialog.



Path
Tap the path elements to change the current folder. The default folder is defined in Default path for all file operations.

You can save the data in a local folder on the instrument, to an external storage device (usually a USB flash drive), or to a folder on a connected network drive. The path list provides all available drives and folders.

On the left, shortcut icons provide access to often used folders.

Toolbar
The toolbar on the top provides various functions for file management.

File Name
The file name to be loaded or stored to. Double-tap the file name, or tap the keyboard icon to enter the file name using the online keyboard.

The default file name for new files is defined in the "Autonaming" tab, see Chapter 4.8.1, "Autonaming", on page 104.

File Type
The file extension of the file to be loaded or stored to.

Save, Select
Selects the specified file for the open or save operation and closes the dialog box.

Cancel
Closes the dialog box without selecting a file.

13 Protocol analysis

Using the serial protocol options for the R&S RTO6, you can analyze various serial protocols.

- Basics of protocol analysis....484
- I²C (option R&S RTO6-K510)....494
- SPI bus (option R&S RTO6-K510)....509
- UART/RS-232/RS-422/RS-485 (option R&S RTO6-K510)....520
- Audio signals (option R&S RTO6-K510)....529
- Custom: Manchester / NRZ (option R&S RTO6-K510)....547
- CAN and CAN FD (option R&S RTO6-K520)....578
- LIN (option R&S RTO6-K520)....604
- FlexRay (option R&S RTO6-K520)....618
- SENT (option R&S RTO6-K520)....634
- CXPI (option R&S RTO6-K520)....666
- MIL-1553 (option R&S RTO6-K530)....685
- ARINC 429 (option R&S RTO6-K530)....702
- SpaceWire (option R&S RTO6-K530)....714
- Ethernet 10BASE-T and 100BASE-TX (option R&S RTO6-K540)....726
- MDIO (option R&S RTO6-K540)....741
- RFFE (option R&S RTO6-K550)....754
- Ethernet 100BASE-T1 (option R&S RTO6-K560)....776
- Ethernet 1000BASE-T1 (option R&S RTO6-K560)....794
- USB (option R&S RTO6-K570)....812
- USB 3.1 (option R&S RTO6-K570)....840
- USBPD (option R&S RTO6-K570)....861
- D-PHY (option R&S RTO6-K580)....874
- M-PHY and USB SSIC (option R&S RTO6-K580 and K570)....891
- 8b/10b (option R&S RTO6-K590)....921
- PCIe (option R&S RTO6-K72)....934
- DDR (option R&S RTO6-K91)....964

13.1 Basics of protocol analysis

Before you can analyze a serial signal, the bus has to be configured according to the protocol and specifics of the signal. The configuration contains:

- Assignment of the data and clock lines to the input channels
- Logical thresholds
- Protocol-specific settings

Serial data can be analyzed in several ways:

- Triggering: You can trigger on various events that are typical for the selected protocol type, for example, on start and stop of messages, or on specified data patterns in the message.

Triggering on a trigger event sequence is not supported, and holdoff settings are not available.

- Protocol decoding: The digitized signal data is displayed on the screen together with the decoded content of the messages in readable form, and the decode results are listed in a table.
- Search on decoded signal data: For most serial protocols, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, because several event types can be combined. Thus, you get the results for the complete acquisition cycle.

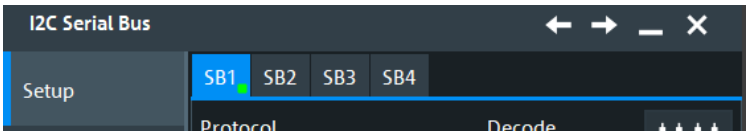
13.1.1 Setup - general settings

For all protocols, configuration starts with the selection of the serial bus and the protocol.

Table 13-1: Configuration settings are protocol-specific. They are described in the related chapters:

K510 Low speed serial buses	I²C	Chapter 13.2.2, "I²C configuration", on page 496
	Custom: Manchester / NRZ	Chapter 13.6.2, "Custom: Manchester / NRZ configuration", on page 549
	UART	Chapter 13.4.2.1, "UART configuration settings", on page 521
	SPI	Chapter 13.3.2, "SPI configuration", on page 510
	Audio	Chapter 13.5.2, "Audio configuration", on page 531
K520 Automotive serial buses low speed	CAN, CAN FD	Chapter 13.7.2, "CAN and CAN FD configuration", on page 579
	LIN	Chapter 13.8.2, "LIN configuration", on page 606
	FlexRay	Chapter 13.9.1.1, "FlexRay configuration", on page 618
	CXPI	Chapter 13.11.2, "CXPI configuration", on page 667
	SENT	Chapter 13.10.2.1, "SENT configuration", on page 639
RTO6-K530 Aerospace buses	MIL-1553	Chapter 13.12.2, "MIL-STD-1553 configuration", on page 688
	ARINC 429	Chapter 13.13.2, "ARINC 429 configuration", on page 703
	SpaceWire	Chapter 13.14.2, "SpaceWire configuration", on page 715

RTO6-K540 Ethernet buses	Ethernet 10BASE-T and 100BASE-TX	Chapter 13.15.2, "Ethernet configuration", on page 727
	MDIO	Chapter 13.16.2, "MDIO configuration", on page 743
RTO6-K550 MIPI low speed buses	MIPI RFFE	Chapter 13.17.2, "RFFE configuration", on page 757
RTO6-K560 Automotive ethernet	Ethernet 100BASE-T1	Chapter 13.18.2, "100BASE-T1 configuration", on page 777
RTO6-K570 USB buses	USB	Chapter 13.20.2, "USB 2.0 configuration", on page 818
	USB 3.1	Chapter 13.21.2, "USB 3.1 configuration", on page 848
	USB PD	Chapter 13.22.2, "USBPD configuration", on page 863
	USB SSIC	Chapter 13.24.2, "M-PHY configuration", on page 893
RTO6-K580 MIPI M-PHY, D-PHY, C-PHY Buses	MIPI D-PHY	Chapter 13.23.2, "D-PHY configuration", on page 875
	MIPI M-PHY	Chapter 13.24, "M-PHY and USB SSIC (option R&S RTO6-K580 and K570)", on page 891
RTO6-K590 PCI express	8b/10b	Chapter 13.25.2, "8b/10b configuration", on page 922
	PCIe	Chapter 13.26.2, "PCIe configuration", on page 937



Make sure that the tab of the correct serial bus is selected.

Protocol
Defines the protocol type of the selected serial bus.

Remote command:
`BUS<m>:TYPE` on page 1670

Decode
Enables the decoding of the selected bus. The signal icon of the bus appears on the signal bar.

Remote command:
`BUS<m>[:STATe]` on page 1670

13.1.2 Autoset for protocols

If you want to perform a quick protocol measurement, you can use the "Autoset" function. It allows you to find the correct instrument settings for the desired protocol. It displays at least one decoded frame after the execution.

Performing a protocol autoset

1. Press [Bus] on the front panel.
2. Select the bus you want to configure.
3. Select the "Setup" tab.
4. Tap the "Protocol" you want to configure.
5. Assign the sources for the input channels.
6. Press the "Autoset" button.

"Autoset" performs the following steps:

- Executes "Autoset" for the horizontal and vertical scale.
- Adjusts the horizontal scale to display at least one frame or packet .
- Executes "Find thresholds" to determine the thresholds.
- If necessary, performs bit rate estimation.
- Sets default protocol trigger (frame start) .
- Turns on the protocol decoder.

Remote command:

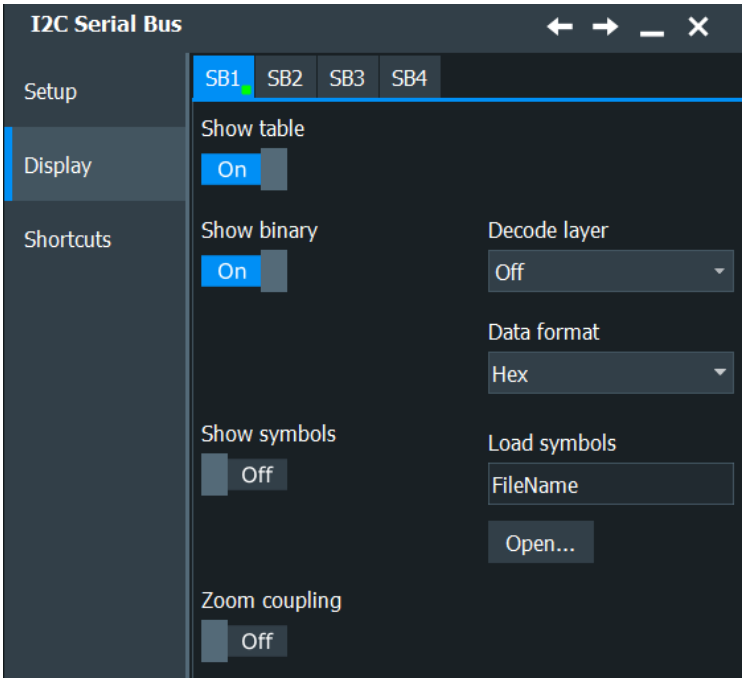
`BUS<m>:FAUToset` on page 1671

There are some limitations for the autoset:

- It is not available, if the digital MSO channels are selected as a source for the waveform.
- Signals with longer idle phases might not find horizontal adjustment.
- On duplex signals, different forward/reverse amplitudes can fail to determine thresholds.

13.1.3 Display

For all protocols, you can select to display the decoded signal as a table and to show the binary signal on the screen.



For some protocols, the result table provides a button to show the details of the selected frame.

Label Lists

For all protocols using ID or address identification, it is possible to create label lists containing addresses or IDs, a symbolic name for each node (symbolic label), and some protocol-specific information.

You can load label lists, and activate its usage for decoding. As a result, an additional "Label" column appears in the "Decode results" table, containing the symbolic label. The frame captions of the decoded signal show the symbolic label instead of the ID or address values. Hence it is easy to identify the messages of the different bus nodes.

You can also use the label list to trigger on an identifier or address. Instead of entering the value, you select the name, which is defined in the label list.

Show table
Opens a table with decoded data of the serial signal. The function requires the option for the analyzed protocol.

Decode results are protocol-specific. They are described in the "Decode Results" chapter of each protocol.

Remote command:
`BUS<m>:RESult` on page 1671

Show binary
For each configured line, the binary signal is also displayed.

Data format
Sets the data format for the values displayed in the decode table and in the combs of the decoded signal.

Remote command:
`BUS<m>:FORMat` on page 1672

Show symbols
Activates the symbols list to be used for decoding. The "Label" appear in the "Decode results" table and in the frame captions of the decoded signal.

Remote command:
`BUS<m>:SYMBols` on page 1672

Load symbols
Selects and loads a label list file. Available file formats are PTT, CSV, DBC (CAN only), and XML (FIBEX files, FlexRay only).

Label lists are protocol-specific. Their contents are described in the corresponding protocol chapters.

Remote command:
`BUS<m>:NEWList` on page 1672

Zoom coupling
If enabled, the decode zoom and result table are synchronized. If you select a row in the result table, this result is shown in the decode zoom.

With "Zoom window", you can also select the zoom used for coupling.

For an introduction to zoom settings and setup, see:
- Chapter 7.1.2, "Zoom settings", on page 241
- Chapter 7.1.3, "Zooming for details", on page 245

Remote command:
`BUS<m>:ZCOupling` on page 1672

13.1.4 Content and format of the PTT file

Label lists are stored as PTT (protocol translation table) files. The PTT file format is an extension of the CSV format (comma-separated values). You can edit it with standard editors, for example, with MS Excel or a text editor.

The PTT file has three types of lines:
- Comment lines begin with a hash character #. A hash character at any other position in the line is treated like a standard character.
- Command lines begin with a commercial at character @. An @ character at any other position in the line is treated like a standard character.
- Standard lines are the lines that not qualify as comment or command lines. They build the core of the label list.

Command lines

Command lines define the version of the PTT file and the protocol name:

- @FILE_VERSION: must appear exactly once in the file
- @PROTOCOL_NAME: must appear at least once in the file. Thus, one file can contain several label lists for different protocols.

```
# --- Start of PTT file
@FILE_VERSION  = 1.0
@PROTOCOL_NAME = i2c
[... Label list for I2C]
@PROTOCOL_NAME = can
[... Label list for CAN]
# --- End of PTT file
```

Standard lines

Standard lines define the contents of the label list. The rules for standard lines follow the csv convention, they are:

- Values are separated by commas
- Space characters following a delimiter are ignored
- Values with a special character (comma, newline, or double quote) must be enclosed in double quotes
- Text in double quotes must be escaped by double quote characters

The format of the numeric value is indicated by a suffix. The following formats are supported:

Format	Suffix	Example
Decimal	<empty> d	`106, DeviceName` `106d, DeviceName`
Hexadecimal	h	`6Ah, DeviceName` or prefix: `0x6A, DeviceName`
Octal	o	`152o, DeviceName`
Binary	b	`01101010b, DeviceName`

The maximum supported word size for (unsigned) integers is 64 bits.

```
# --- Start of PTT file
@FILE_VERSION  = 1.0
@PROTOCOL_NAME = i2c
#    Following two lines are equal:
7,01h,Temperature
7,01h, Temperature
#    A comma must be enclosed in double quotes:
7,01h,"Temperature, Pressure, and Volume"
#    A double quote must also be enclosed in double quotes:
7,7Fh,"Highspeed ""Master"" 01"
```

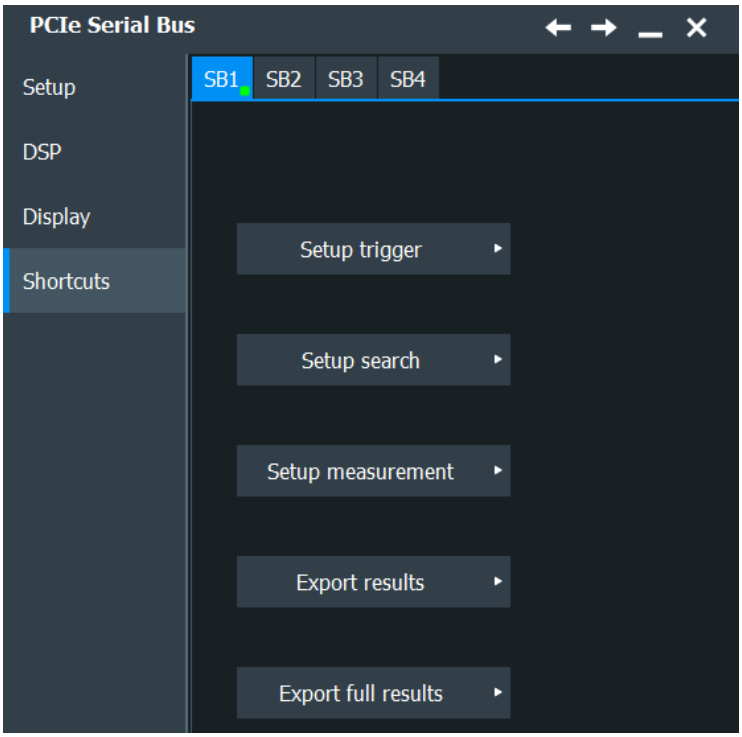
```
#    Following lines yield the same result:
7d,0x11,Pressure
7h,11h,Pressure
0x7,17d,Pressure
7,17,Pressure
```

13.1.5 Shortcuts

The "Shortcuts" tab gives quick access to other dialogs with protocol relevant settings. The availability depends on the functions supported by the protocol.

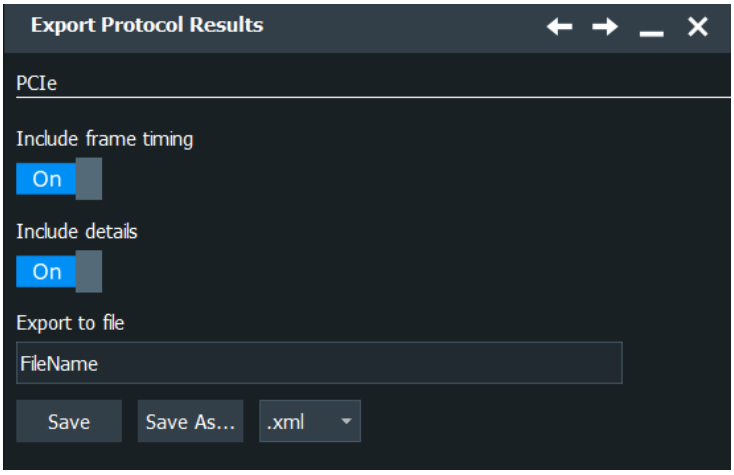
If you access a dialog from the "Shortcuts" tab, some settings are already predefined.

- "Trigger" dialog: you can trigger on various events that are typical for the selected protocol type, for example, on start and stop of messages, or on specified data patterns in the message.
- "Search" dialog: here you can configure protocol-specific search settings.
- In the "Measurements" dialog, you can perform additional measurements for some protocols. See Chapter 8.2.11, "Protocol measurements (option R&S RTO6-K500)", on page 349.
- "Export reults": in the "Save Results" dialog, you can select the decode results you want to export, the file format, and the delimiter.
  For details, see Chapter 12.2.4, "Results", on page 467.
- "Export full results": in this dialog, you can select the decode results you want to export and the protocol-specific details to be included.



13.1.6 Export protocol results

In the "Export Protocol Results" dialog, you can export the results and all details of the selected protocol.



Include frame timing
Includes the frame timing.

Remote command:
`BUS<m>:EXPResult:TIME` on page 1673

Include details
Includes the detailed results for all frames.

Remote command:
`BUS<m>:EXPResult:DETail` on page 1673

Export to file
Enter the filename to save the results to. Double-tap the filename to open the file selection dialog box.

"Save"	Saves the selected results to the indicated file.
"Save As..."	Opens the file selection dialog box and saves the selected results to the selected file. See also Chapter 12.5, "File selection dialog", on page 482

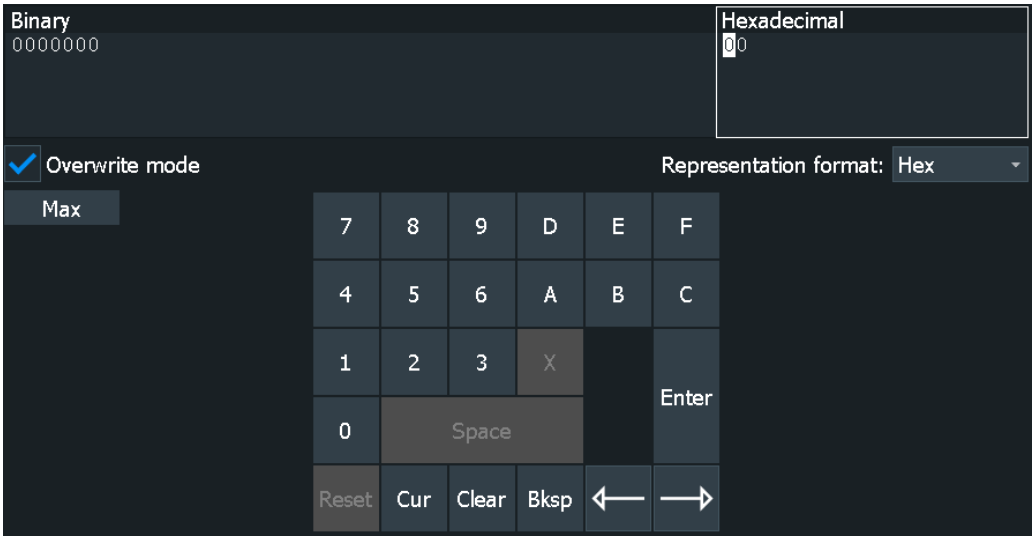
File format Selects the file format.

- ".csv": the values are saved in a file.
  You can select the value delimiter and the list separator symbol in the "CSV Export" dialog, see Chapter 4.8.2, "CSV export", on page 107.
  Tip for using MS Excel: It is recommended that you use the semicolon as csv decimal symbol. When you open the file with MS Excel, use "File > Open" and follow the wizard to set the separators correctly, or set the separator settings with "Tools > Options > International".
- ".html": the results are saved as web page for display in a browser.
- ".xml": the results are saved in an xml compatible file format.
- ".py": the values are saved in a python compatible file format.

Remote command:
`BUS<m>:EXPResult:SAVE` on page 1673

13.1.7 Bit pattern editor

If you want to enter a specified address or data pattern, the bit pattern editor helps you to enter the pattern in various formats - decimal, hexadecimal, octal, binary and ASCII.



The editor displays the pattern in two columns. The left column always shows binary data. For the right column, you can select the format, the default depends on the data specifics. You can edit data in the left or right column. The keypad adapts itself to the column format, only keys appropriate to the format are enabled.

The data is grouped and converted in bit groups. The size of a bit group depends on the address or data specifics and is set by the instrument. Groups are automatically separated by blanks. The maximum size of a bit group is 64 bit, the most common group size is 1 byte.

"Overwrite mode": If disabled, the data behind the new digit is shifted to the right. Bit groups are rearranged automatically.

Format-specific information:

- Unsigned: Decimal data format without sign. It is available for I²C, SPI, UART, CAN, LIN and FlexRay protocols. If you enter a decimal number that is too large for the defined bit group, the number is truncated and a message appears. X (do not care) in the decimal column sets all binary digits of the bit group to X.
- Signed: Signed decimal format, available for audio protocols. The first bit represents the sign. You can use the 2's complement or 1's complement format.
- Binary: 0, 1 and X (do not care) is allowed.
- Hex: most common format in the right column.
- Octal: Each digit represents 3 bit.
- ASCII: In the ASCII column, "X" is the character X. The binary X (do not care) is not allowed. If an X is included in the binary value in the left column, the ASCII column displays "§" to indicate that the value is not defined.

Where applicable, frequently used values are provided in a "Predefined values" list below the pattern table, for example, reserved end words of data packets in the UART protocol.

13.2 I²C (option R&S RTO6-K510)

The Inter-Integrated Circuit is a simple, low-bandwidth, low-speed protocol used for communication between on-board devices.

- The I²C protocol 494
- I²C configuration 496
- I²C trigger 499
- I²C decode results 505
- Search on decoded I²C data 506

13.2.1 The I²C protocol

This chapter provides an overview of protocol characteristics, data format, address types and trigger possibilities. For detailed information, read the "I2C-bus specification and user manual" available on the NXP manuals webpage at http://www.nxp.com/.

I²C characteristics

Main characteristics of I²C are:

- Two-wire design: serial clock (SCL) and serial data (SDA) lines
- Master-slave communication: the master generates the clock and addresses the slaves. Slaves receive the address and the clock. Both master and slaves can transmit and receive data.

- Addressing scheme: each slave device is addressable by a unique address. Multiple slave devices can be linked together and can be addressed by the same master.
- Read/write bit: specifies if the master reads (=1) or writes (=0) the data.
- Acknowledge: takes place after every byte. The receiver of the address or data sends the acknowledge bit to the transmitter.

The R&S RTO6 supports all operating speed modes: high-speed, fast mode plus, fast mode, and standard mode.

Data transfer

The format of a simple I²C message (frame) with 7-bit addressing consists of the following parts:

- Start condition: a falling slope on SDA while SCL is high
- 7-bit address of the slave device that either is written to or read from
- R/W bit: specifies if the data is written to or read from the slave
- ACKnowledge bits: is issued by the receiver of the previous byte if the transfer was successful
  Exception: At read access, the master terminates the data transmission with a NACK bit after the last byte.
- Data: several data bytes with an ACK bit after every byte
- Stop condition: a rising slope on SDA while SCL is high

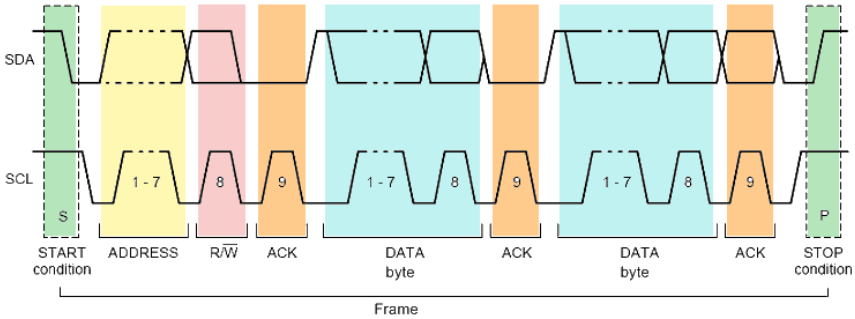


Figure 13-1: I2C writes access with 7-bit address

Address types: 7-bit and 10-bit

Slave addresses can be 7 bits or 10 bits long. A 7-bit address requires 1 byte, 7 bits for the address followed by the R/W bit.

A 10-bit address for write access requires 2 bytes: the first byte starts with the reserved sequence 11110, followed by the two MSB of the address and the write bit. The second byte contains the remaining 8 LSB of the address. The slave acknowledges each address byte.

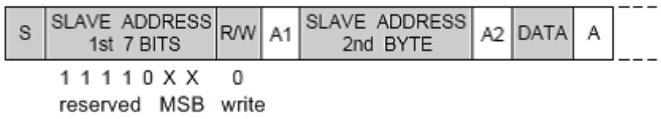


Figure 13-2: 10-bit address, write access

A 10-bit address for read access requires 3 bytes. The first 2 bytes are identical to the write access address. The third byte repeats the address bits of the first byte and sets the read bit.

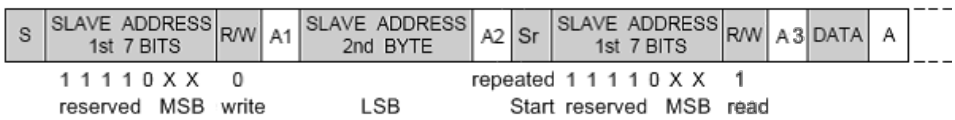


Figure 13-3: 10-bit address, read access

Trigger

The R&S RTO6 can trigger on various parts of I²C messages. The data and clock lines must be connected to the input channels, triggering on math and reference waveforms is not possible.

You can trigger on:

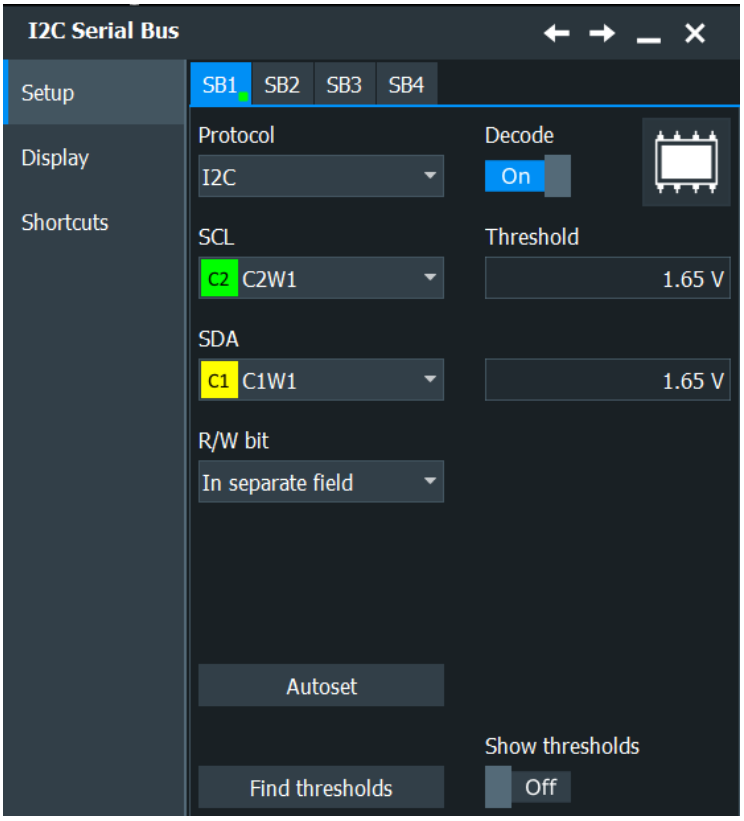
- Start or stop condition
- Repeated start condition
- Transfer direction (read or write)
- Bytes with missing acknowledge bit
- Specific slave address or address range
- Specific data pattern in the message

13.2.2 I²C configuration

13.2.2.1 I²C configuration settings

Access: [Bus] > "I2C" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

SDA, SCL

Set the waveforms of the data line (SDA) and clock line (SCL).

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Do not combine a reference waveform with channel or math waveform because the time correlation of these waveforms might differ.

Alternatively, digital channels can be used if MSO option R&S RTO6-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Remote command:

`BUS<m>:I2C:SDA:SOURce` on page 1676

`BUS<m>:I2C:SCL:SOURce` on page 1676

R/W bit

Defines if the R/W bit is considered separately or as part of the address. The setting affects the Adress Type of the trigger conditions.

Remote command:

`BUS<m>:I2C:RWBit` on page 1678

Threshold
Sets the threshold value for digitization of signals for each line. If the signal value on the line is higher than the threshold, the signal state is high (1 or true for the Boolean logic). Otherwise, the signal state is considered low (0 or false) if the signal value is below the threshold.

There are several ways to set the threshold:
- "Threshold"
  Enter the value directly in the field.
- "Autoset"
  Starts software algorithms for determining the signal threshold levels and bitrate.
- "Find thresholds"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Note: If the sources are digital channels, the same threshold values are used for the parallel and the serial buses. You can set the thresholds either in the parallel bus configuration or in the serial bus configuration.

Remote command:
`BUS<m>:I2C:SCL:THReshold` on page 1677
`BUS<m>:I2C:SDA:THReshold` on page 1677
`BUS<m>:I2C:TECHnology` on page 1677
`BUS<m>:SETReflevels` on page 1671
`BUS<m>:FAUToset` on page 1671

Show thresholds
If enabled, the threshold levels are displayed in the diagram.

Remote command:
`BUS<m>:THReshold` on page 1671

13.2.2.2 Display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

I2C Specific Label Settings

Label lists are protocol-specific. Label lists for I²C are available in CSV and PTT format.

An I²C label file contains three values for each address:
- Address type, 7-bit or 10-bit long
- Address value
- Symbolic label: name of the address, specifying its function in the bus network.

Example: I²C PTT file

```
# ---------------------------------------------------------------------------
@FILE_VERSION  = 1.00
@PROTOCOL_NAME = i2c
```

```
# ---------------------------------------------------------------------------
# Labels for I2C protocol
#   Column order: Identifier type, Identifier value, Label
# ---------------------------------------------------------------------------
7,0x1E,Voltage
7,38h,Pressure
7,2Ah,Temperature
7,16h,Speed
7,118,Acceleration
7,07h,HighSpeed_Master_0x3
7,51h,EEPROM
10,3A2h,DeviceSetup
10,1A3h,GatewayStatus
10,06Eh,LeftSensor
# ---------------------------------------------------------------------------
```

13.2.2.3 Configuring I²C protocol

The configuration of the I²C is simple - assign the two lines to input channels, and set the thresholds.

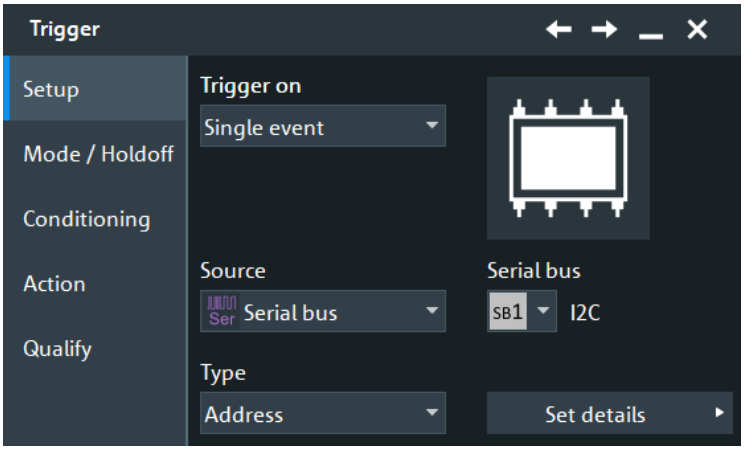
For details on configuration settings, see Chapter 13.2.2, "I²C configuration", on page 496.

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. Select the "Setup" tab.
4. Tap the "Protocol" button and select the protocol: "I2C".
5. Tap the "SDA" button, and select the waveform of the data line.
6. Tap the "SCL" button, and select the waveform of the clock line.
7. Set the logical thresholds.
8. Enable "Decode".

13.2.3 I²C trigger

13.2.3.1 I²C trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Serial bus

Selects the serial bus to be triggered on. Make sure to select the correct bus before you enter the settings.

To trigger on a serial bus, the signals sources must be channel signals. If the data or clock source is a math or reference waveform, you cannot trigger on that bus.

Remote command:

`TRIGger<m>:SOURce:SBSelect` on page 1675

Trigger type

Selects the trigger type for I²C analysis.

Some trigger types have additional settings that can be defined. In this case, the "Define trigger details" button appears next to the "Type" function. Open it for a detailed definition of the trigger conditions.

Remote command:

`TRIGger<m>:I2C:TYPE` on page 1679

Start ← Trigger type

Sets the trigger to the start of the message. The start condition is a falling edge on SDA while SCL is high. The trigger instant is the falling edge of the SDA line.

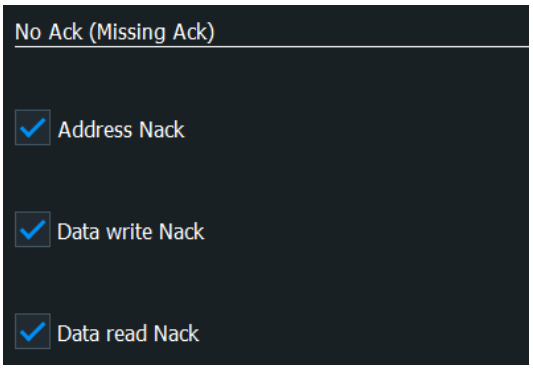
You can change the SDA and SCL lines here if necessary.

Repeated start ← Trigger type
Sets the trigger to a repeated start - when the start condition occurs without previous stop condition. Repeated start conditions occur when a primary exchanges multiple messages with a secondary without releasing the bus.

Stop ← Trigger type
Sets the trigger to the end of the message. The stop condition is a rising slope on SDA while SCL is high.

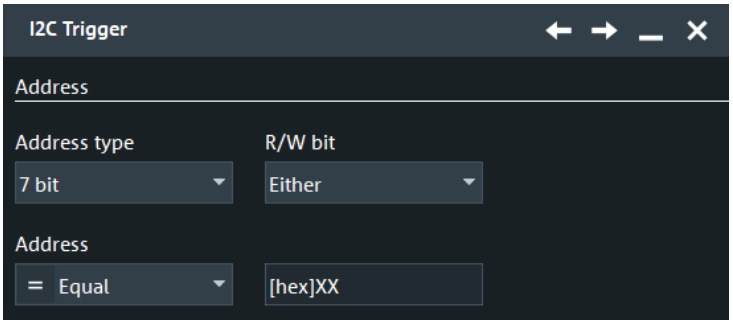
No Ack (Missing Ack) ← Trigger type
Missing acknowledge bit: the instrument triggers if the data line remains HIGH during the clock pulse following a transmitted byte.

You can also localize specific missing acknowledge bits by setting the No Ack conditions.



Address ← Trigger type
Sets the trigger to one specific address condition or a combination of address conditions. The trigger time is the falling clock edge of the acknowledge bit after the address. You can specify:

- Adress Type
- "Address" on page 503
- R/W bit

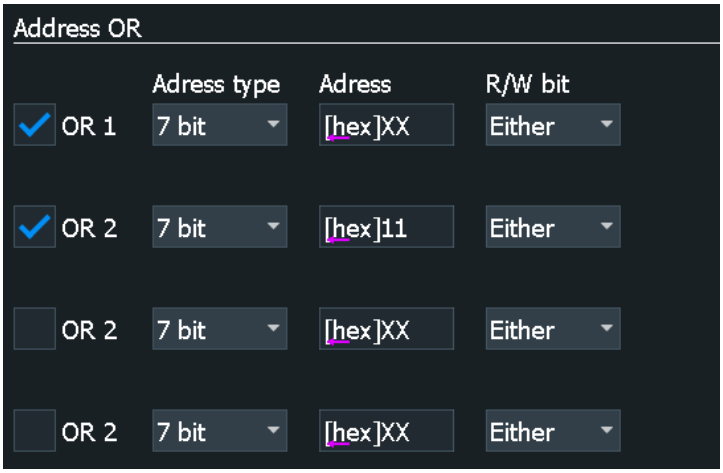


Address OR ← Trigger type
Triggers on one to four address conditions. Each enabled condition requires an exact address. You can specify:

For each condition to be used, select "Monitor"

- "OR x" on page 504

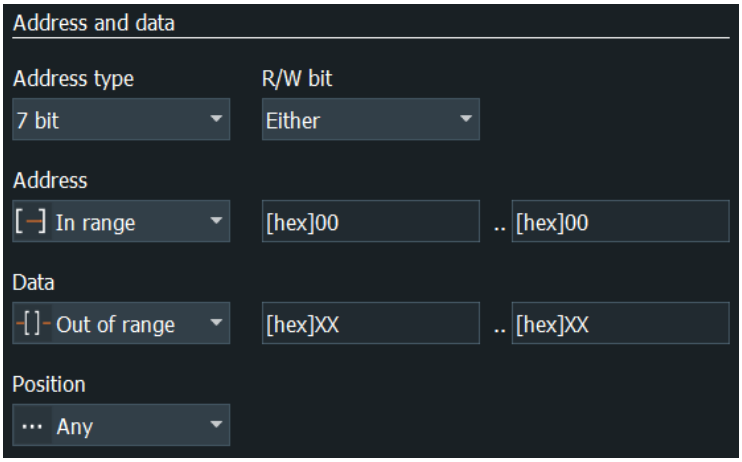
- Adress Type
- "Address" on page 503
- R/W bit



Address and data ← Trigger type
Sets the trigger to a combination of address and data condition. You can specify:

- Adress Type
- R/W bit
- "Address" on page 503
- "Position" on page 504
- "Data" on page 504

The address conditions are the same as for the "Address" trigger type and "Data" on page 504.



No Ack conditions
Selects which missing acknowledge bits is detected if the trigger type is set to "Missing Ack".

"Address Nack"
No secondary recognizes the address.

"Data write Nack"
The addressed secondary does not accept the data.

"Data read Nack"
Marks the end of the read process when the primary reads data from the secondary. This Nack is sent according to the protocol definition, it is not an error.

Remote command:
`TRIGger<m>:I2C:ADNack` on page 1680
`TRIGger<m>:I2C:DWNack` on page 1680
`TRIGger<m>:I2C:DRNack` on page 1680

Adress Type
Sets the address length to be triggered on: 7 bit, 7+1 bit, or 10 bit. Available settings depend on the R/W bit setting of the bus configuration.

For "7 bit" and "10 bit", enter the address bits in the Address field, and use the "R/W bit" on page 503 field to select the transfer direction.

For "7+1 bit", enter the seven address bits and also the R/W bit in the "Address" field.

If the trigger type is "Address + data", you can set the address type "Any" to trigger on data only, regardless of the address.

Remote command:
`TRIGger<m>:I2C:AMODe` on page 1680
`TRIGger<m>:I2C:ADOR<n>:ADRType` on page 1682

Address
The trigger address setup consists of a comparison condition and one or two address patterns.

Defines the bit pattern of the secondary device address. The length of the entry is adjusted to the selected address type.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Condition" Sets the comparison condition to a specific value or a range.

"Min" Specifies the value or sets the start value of a range.

"Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:I2C:ADDRess` on page 1681
`TRIGger<m>:I2C:ADDTo` on page 1681
`TRIGger<m>:I2C:ADOR<n>[:VALue]` on page 1682

R/W bit
Toggles the trigger condition between read and write access of the primary. Select "Either" if the transfer direction is not relevant for the trigger condition.

Remote command:
`TRIGger<m>:I2C:ACCess` on page 1679
`TRIGger<m>:I2C:ADOR<n>:RWBit` on page 1682

OR x
Enables each selected condition, for "Trigger Type" = "Address OR".

Remote command:
`TRIGger<m>:I2C:ADOR<n>:ENABle` on page 1681

Position
Sets the number of data bytes to be skipped after the address. You can define an exact position, or a position range. The index 0 is associated with the first data byte.

"Condition"	Sets the comparison condition to a specific value or a range.
"Index min"	Specifies the value or sets the start value of a range.
" Index max"	Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:I2C:DPOPerator` on page 1683
`TRIGger<m>:I2C:DPOSition` on page 1683
`TRIGger<m>:I2C:DPTO` on page 1683

Data
Specifies the trigger conditions for the data bit pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Condition"	Sets the comparison condition to a specific value or a range.
"Value min "	Specifies the value or sets the start value of a range. Enter the bytes in msb first bit order. The maximum pattern length is 64 bit. Waveform data is compared with the pattern byte-by-byte.
"Value max "	Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:I2C:DCONdition` on page 1683
`TRIGger<m>:I2C:DMIN` on page 1684
`TRIGger<m>:I2C:DMAX` on page 1684

13.2.3.2 Triggering on I²C signals

Prerequisites: An I²C bus is configured, see Chapter 13.2.2.3, "Configuring I²C protocol", on page 499.

1. Press the [Bus] key and select the "Shortcuts" tab.
2. Press "Setup trigger".

3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to I²C.
5. Select the "Trigger type".
6. For more complex trigger types, enter the address and/or data conditions: address, acknowledge bits, R/W bit, and data pattern.
   For details, see Chapter 13.2.3, "I²C trigger", on page 499

13.2.4 I²C decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Serial Bus" dialog > "Setup" tab, enable "Decode".
2. In the "Serial Bus" dialog > "Display" tab, select additional result display settings: "Show table" and "Show binary".
3. If required, enable "Zoom coupling"

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

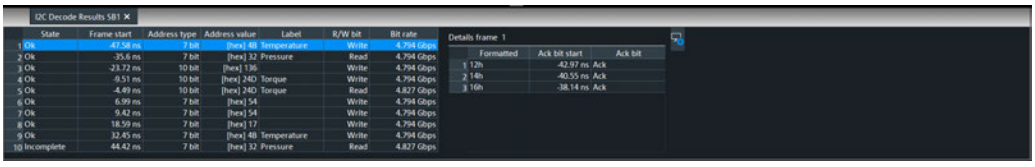


Table 13-2: Content of the Details frame table

Column	Description
State	Overall state of the frame. "Insuffcient waveform length" indicates that the frame is not completely contained in the acquisition. Change the horizontal scale, or move the reference point to the left to get a longer acquisition.
Frame start	Time of frame start

Column	Description
Address type	Address length, 7 bit or 10 bit
Address value (hex)	Hexadecimal value of the address
Label	Symbolic label name defined in the label list
R/W bit	Value of the R/W bit
Bit rate	Value of the bit rate

Table 13-3: Content of the Decode results table

Column	Description
Formatted	Value of all data bytes of the frame
Ack bit start	Start time of the acknowledge bit
Ack bit	Value of the address acknowledge bit

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands are described in Chapter 24.17.3.3, "Decode results", on page 1684.

13.2.5 Search on decoded I²C data

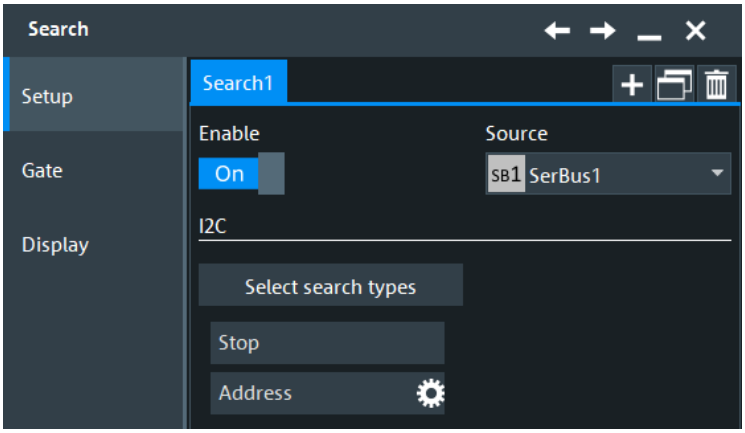
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 11, "Search functions", on page 419.

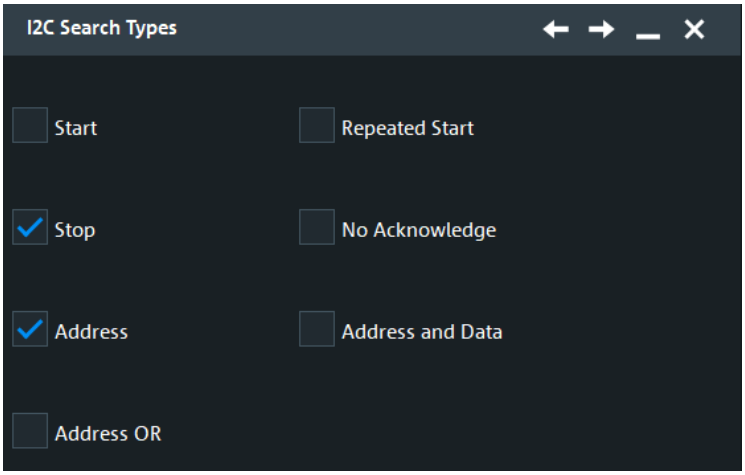
13.2.5.1 I²C search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"



Select search types

Enable the events to be searched for. Unlike triggering, where you can trigger only on one defined event, you can search for various different events in one search.



"Start"	Searches for the start of the message. The start condition is a falling edge on SDA while SCL is high. The event is the falling edge of the SDA line. You can change the SDA and SCL lines here if necessary.
"Repeated start"	Searches for a start condition without previous stop condition. Repeated start conditions occur when a master exchanges multiple messages with a slave without releasing the bus.
"Stop"	Searches for the end of the message. The stop condition is a rising slope on SDA while SCL is high.
"No ACK (Missing ACK)"	Searches for a missing acknowledge bit: an event is found if the data line remains HIGH during the clock pulse following a transmitted byte.
"Address"	Searches for one specific address condition or a combination of address conditions. The event is the falling edge of the acknowledge bit after the address.
"Address OR"	Searches for one to four address conditions.

"Address and data" Searches for a combination of address and data conditions.

Remote command:
`SEARch:TRIGger:I2C:SCONdition` on page 1693
`SEARch:TRIGger:I2C:RCONdition` on page 1693
`SEARch:TRIGger:I2C:STCNdition` on page 1694
`SEARch:TRIGger:I2C:NACKnowledge` on page 1694
`SEARch:TRIGger:I2C:SADDress` on page 1694
`SEARch:TRIGger:I2C:ADOR` on page 1695
`SEARch:TRIGger:I2C:ADData` on page 1695

Address setup: Condition, Type, Addr. min, Addr. max, R/W bit
The address setup consists of the condition, type, R/W bit and one or two address patterns.

The address setup settings are the same as in the I2C trigger setup, see "Address" on page 503.

Remote command:
`SEARch:TRIGger:I2C:ACONdition` on page 1695
`SEARch:TRIGger:I2C:ADDRess` on page 1696
`SEARch:TRIGger:I2C:ADDTo` on page 1696
`SEARch:TRIGger:I2C:AMODe` on page 1696
`SEARch:TRIGger:I2C:ACCess` on page 1697

Data setup: Condition, Position, Index min, Index max, Value min, Value max
The data setup consists of the condition, position, and one or two index/value patterns.

The data setup settings are the same as in the I2C trigger setup, see "Data" on page 504.

Remote command:
`SEARch:TRIGger:I2C:DPOPerator` on page 1698
`SEARch:TRIGger:I2C:DPOSition` on page 1699
`SEARch:TRIGger:I2C:DPTO` on page 1699
`SEARch:TRIGger:I2C:DCONdition` on page 1699
`SEARch:TRIGger:I2C:DMIN` on page 1700
`SEARch:TRIGger:I2C:DMAX` on page 1700

Address OR setup: , OR x Address type, Address, R/W bit
The address OR setup consists of the monitor, address type, address and an R/W bit.

The address OR setup settings are the same as in the I2C trigger setup, see Chapter 13.2.3.1, "I²C trigger settings", on page 499.

Remote command:
`SEARch:TRIGger:I2C:ADDO<m>:ENABle` on page 1697
`SEARch:TRIGger:I2C:ADDO<m>:ADRType` on page 1697
`SEARch:TRIGger:I2C:ADDO<m>[:VALue]` on page 1698
`SEARch:TRIGger:I2C:ADDO<m>:RWBit` on page 1698

No ACK setup: Addr/W/R NACK
The no ACK setup consists of the Addr/W/R NACK.

The no ACK setup settings are the same as in the I2C trigger setup, see "No Ack conditions" on page 502.

Remote command:
`SEARch:TRIGger:I2C:DRNack` on page 1700
`SEARch:TRIGger:I2C:DWNack` on page 1701
`SEARch:TRIGger:I2C:NACKnowledge` on page 1694

13.2.5.2 I²C search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

The remote commands for the search reults are described in Chapter 24.17.3.5, "I²C search results", on page 1701.

13.3 SPI bus (option R&S RTO6-K510)

- The SPI protocol 509
- SPI configuration 510
- SPI trigger 515
- SPI decode results 517
- Search on decoded SPI data 518

13.3.1 The SPI protocol

A 4-channel instrument is required for full support of the SPI protocol, or the MSO option R&S RTO6-B1.

The Serial Peripheral Interface SPI is used for communication with slow peripheral devices, in particular, for transmission of data streams.

Main characteristics of SPI are:

- Master-slave communication
- No device addressing; The slave is accessed by a chip select, or slave select line.
- No acknowledgement mechanism to confirm receipt of data
- Duplex capability

Most SPI buses have four lines, two data and two control lines:

- Clock line to all slaves (SCLK)
- Slave Select or Chip Select line (SS or CS)
- Master data output, slave data input (MOSI or SDI)
- Master data input, slave data output (MISO or SDO)

When the master generates a clock and selects a slave device, data may be transferred in either or both directions simultaneously.

Figure 13-4: Simple configuration of SPI bus

The data bits of a message are grouped by following criteria:

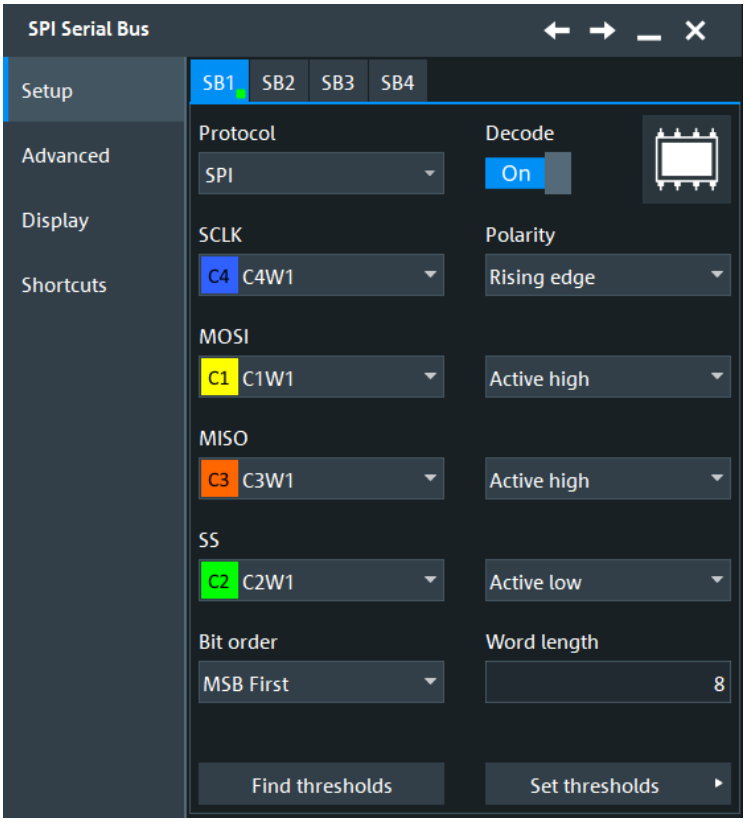
- A word contains a number of successive bits. The word length is defined in the protocol configuration.
- A frame contains a number of successive words, at least one word.

13.3.2 SPI configuration

13.3.2.1 SPI configuration settings

Access: [Bus] > "SPI" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

SCLK
Defines the settings for the clock line.

SCLK source ← SCLK
Sets the input channel of the clock line. Waveform 1 of channel signals, math waveforms, and reference waveforms can be used for decoding.

Alternatively, digital channels can be used if MSO option R&S RTO6-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Remote command:
`BUS<m>:SPI:SCLK:SOURce` on page 1709

Polarity ← SCLK
Two settings define the clock mode: the clock polarity and the clock phase. Together, they determine the edges of the clock signal on which the data are driven and sampled. A master/slave pair must use the same parameter pair values to communicate.

The clock polarity is "Idle low" (idle = 0) or "Idle high" (idle = 1).

The clock phase defines the slope. It selects if data is stored with the rising or falling slope of the clock. The slope marks the begin of a new bit.

SS, MISO, MOSI
Configures the Slave Select, MISO and MOSI lines.

Source ← SS, MISO, MOSI
Sets the input channel of the selected line.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Do not combine a reference waveform with channel or math waveform because the time correlation of these waveforms might differ.

Alternatively, digital channels can be used if MSO option R&S RTO6-B1 is installed. Digital and analog channels cannot be used at the same time.

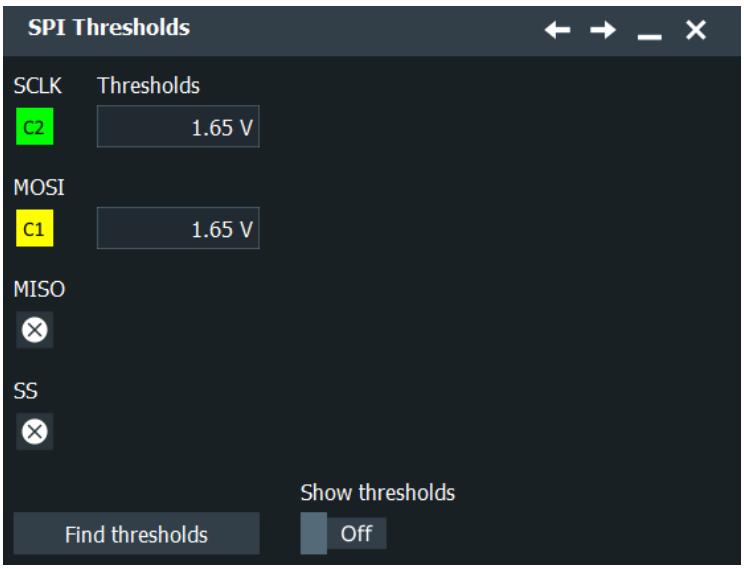
For triggering on a serial bus, analog or digital channel sources are required.

Remote command:
`BUS<m>:SPI:SSELect:SOURce` on page 1709
`BUS<m>:SPI:MISO:SOURce` on page 1710
`BUS<m>:SPI:MOSI:SOURce` on page 1711

Polarity ← SS, MISO, MOSI
Selects whether transmitted data or the slave select signal is high active (high = 1) or low active (low = 1).

Remote command:
`BUS<m>:SPI:SSELect:POLarity` on page 1710
`BUS<m>:SPI:MISO:POLarity` on page 1710
`BUS<m>:SPI:MOSI:POLarity` on page 1711

Set Thresholds
Opens a dialog, where you can set the tresholds for digitization of signals for each line. If the signal value on the line is higher than the threshold, the signal state is high. Otherwise, the signal state is considered low if the signal value is below the threshold. The interpretation of HIGH and LOW is defined by the Polarity.



There are several ways to set the threshold:

- "Threshold"
  Enter the value directly in the field.
- "Autoset"
  Starts software algorithms for determining the signal threshold levels and bitrate.
- "Find thresholds"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Note: If the sources are digital channels, the same threshold values are used for the parallel and the serial buses. You can set the thresholds either in the parallel bus configuration or in the serial bus configuration.

Remote command:
`BUS<m>:SPI:SCLK:THReshold` on page 1711
`BUS<m>:SPI:MISO:THReshold` on page 1711
`BUS<m>:SPI:MOSI:THReshold` on page 1711
`BUS<m>:SPI:SSELect:THReshold` on page 1711
`BUS<m>:SETReflevels` on page 1671
`BUS<m>:FAUToset` on page 1671

Bit order
Defines if the data of the messages starts with msb (most significant bit) or lsb (least significant bit). The display of the decoded signal considers this setting, results are displayed in the specified order.

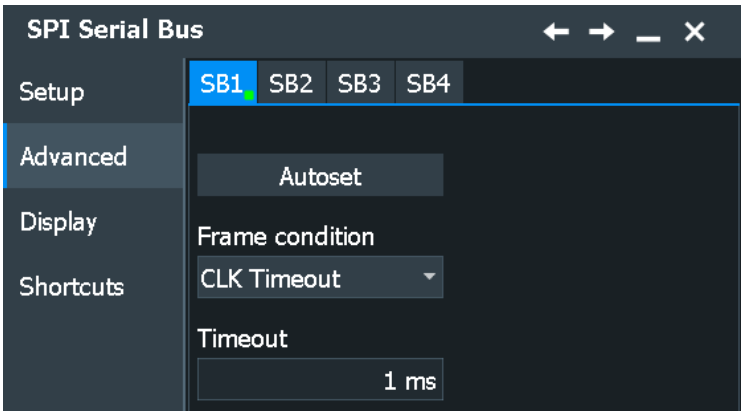
Remote command:
`BUS<m>:SPI:BORDer` on page 1708

Word length
Sets the number of bits in a word. The maximum length is 32 bit.

Remote command:
`BUS<m>:SPI:WSIZe` on page 1709

13.3.2.2 Advanced settings

Access: [Bus] > "Advanced" tab.



Autoset
Starts software algorithms for determining the signal threshold levels and bitrate.

Remote command:
`BUS<m>:FAUToset` on page 1671

Frame condition
Defines the start of a frame. A frame contains a number of successive words, at least one word.

"SS"	Start and end of the frame is defined by the active state of the slave select signal.
"CLK timeout"	Defines a timeout on the clock line SCLK as limiter between two frames. The timeout condition is used for SPI connections without an SS line. Enter the minimum clock idle time in the field.

Remote command:
`BUS<m>:SPI:FRCondition` on page 1712

Timeout
Sets the minimum clock idle time if a timeout on the clock line SCLK is used as limiter between two frames.

See also: "Frame condition" on page 514.

Remote command:
`BUS<m>:SPI:TIMeout` on page 1712

13.3.2.3 Configuring SPI signals

For configuration, assign the lines to the input channels, and define the active states and the logical thresholds.

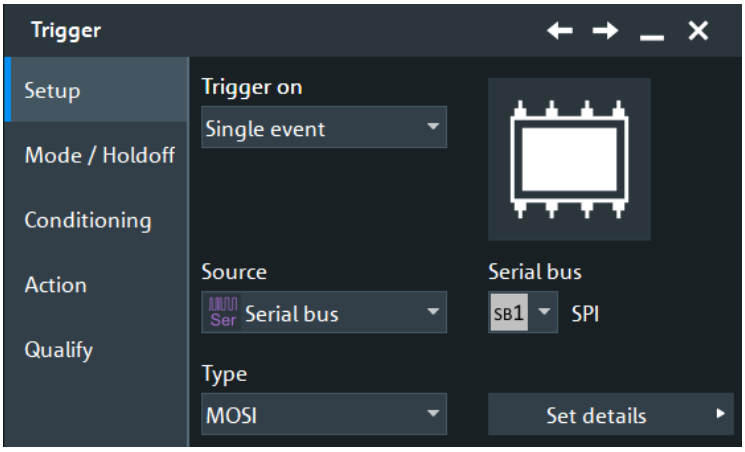
For details on configuration settings, see Chapter 13.3.2, "SPI configuration", on page 510.

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. Tap the "Protocol" button and select the protocol: "SPI".
4. Tap the "SCLK Source" button, and select the waveform of the clock line.
5. Set the polarity (clock mode) for SCLK.
6. For each of the available SS, MISO and MOSI lines, assign the waveform. Define the polarity (active state) of the line.
7. Set the logical thresholds.
8. Set the "Bit order", "Word length", and "Frame condition" according to your signal.
9. Enable "Decode".

13.3.3 SPI trigger

13.3.3.1 SPI trigger

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

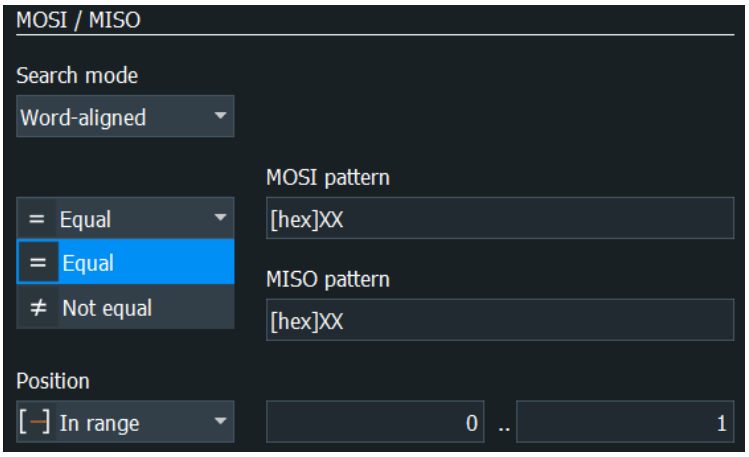
Trigger type
Selects the trigger type for SPI analysis.

"Frame start (SS)"	Trigger on the start of the frame when the slave select signal SS changes to the active state. This trigger type is available if Frame condition is set to "SS".
"Frame start (Timeout)"	Triggers on the start of the frame when the clock idle time exceeds the "Timeout" time. This trigger type is available if Frame condition is set to "CLK timeout".
"MOSI"	Sets the trigger to a specified data pattern expected on the MOSI line. See: "MOSI and MISO data conditions" on page 516.
"MISO"	Sets the trigger to a specified data pattern expected on the MISO line. See: "MOSI and MISO data conditions" on page 516.

"MOSI/MISO" Sets the trigger to specified data patterns expected on the MOSI and MISO lines on the same time.

Remote command:
`TRIGger<m>:SPI:TYPE` on page 1713

MOSI and MISO data conditions
The trigger on MOSI and MISO patterns is defined in the same way:



Condition ← MOSI and MISO data conditions
Selects the operator for the "Data" pattern: "Equal" or "Not equal".

Remote command:
`TRIGger<m>:SPI:FCONdition` on page 1714

MOSI pattern, MISO pattern ← MOSI and MISO data conditions
Specify the data patterns to be found on the MOSI and/or MISO line.

If the trigger type is "MOSI" or "MISO" (one pattern is defined), the maximum pattern length is 256 bit. If the trigger type is "MOSI/MISO", two patterns must be found at the same time. Thus, both patterns must have the same length, and the maximum pattern length of each pattern is 128 bit.

Enter the words in msb first bit order. The starting point of the pattern is defined by Index min, Index max and Search mode.

The bit pattern editor helps you to enter the pattern, see Chapter 13.1.7, "Bit pattern editor", on page 493.

Remote command:
`TRIGger<m>:SPI:MOSipattern` on page 1715
`TRIGger<m>:SPI:MISopattern` on page 1715

Position ← MOSI and MISO data conditions
Operator for the data position. You can define an exact position, a position range, or let the position undefined ("Any").

Remote command:
`TRIGger<m>:SPI:DPOPerator` on page 1714

Index min, Index max ← MOSI and MISO data conditions
The effect of data positioning depends on the Search mode. It sets the number of bits or words before the first word of interest. These offset bits/words are skipped. If the position operator defines a range, the first and the last bit/word of interest are defined. The index 0 is associated with the first data bit or word.

Remote command:
`TRIGger<m>:SPI:DPOSition` on page 1714
`TRIGger<m>:SPI:DPTO` on page 1714

Search mode ← MOSI and MISO data conditions
Defines how the specified data pattern is searched:

"Word-aligned" The pattern is matched only at word boundaries.

"Bit-aligned" Bit-by-bit: the pattern can start at any position in the message.

Remote command:
`TRIGger<m>:SPI:PALignment` on page 1713

13.3.3.2 Triggering on SPI

Prerequisites: A bus is configured for the SPI signal to be analyzed.

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Press "Setup trigger".
3. Select the serial bus that is set to SPI.
4. Select the "Trigger type".
5. For more complex trigger types, enter the data pattern conditions
   For details, see Chapter 13.3.3, "SPI trigger", on page 515

13.3.4 SPI decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

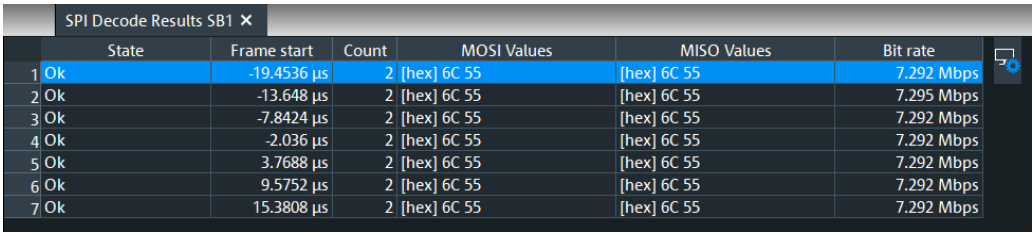
For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The "Decode results" table contains information about all decoded frames.



SPI Decode Results SB1

	State	Frame start	Count	MOSI Values	MISO Values	Bit rate
1	Ok	-19.4536 µs	2	[hex] 6C 55	[hex] 6C 55	7.292 Mbps
2	Ok	-13.648 µs	2	[hex] 6C 55	[hex] 6C 55	7.295 Mbps
3	Ok	-7.8424 µs	2	[hex] 6C 55	[hex] 6C 55	7.292 Mbps
4	Ok	-2.036 µs	2	[hex] 6C 55	[hex] 6C 55	7.292 Mbps
5	Ok	3.7688 µs	2	[hex] 6C 55	[hex] 6C 55	7.292 Mbps
6	Ok	9.5752 µs	2	[hex] 6C 55	[hex] 6C 55	7.292 Mbps
7	Ok	15.3808 µs	2	[hex] 6C 55	[hex] 6C 55	7.292 Mbps

Table 13-4: Content of the "Decode results" table

Column	Description
State	Overall state of the frame
Frame start	Times of frame start and frame end
Count	Number of words in the frame
MOSI values	Value of the MOSI data words. The data format is selected below the table.
MISO values	Value of the MISO data words. The data format is selected below the table.
Bit rate	Value of the bit rate

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands to retrieve decode results are described in Chapter 24.17.4.3, "SPI decode results", on page 1715.

13.3.5 Search on decoded SPI data

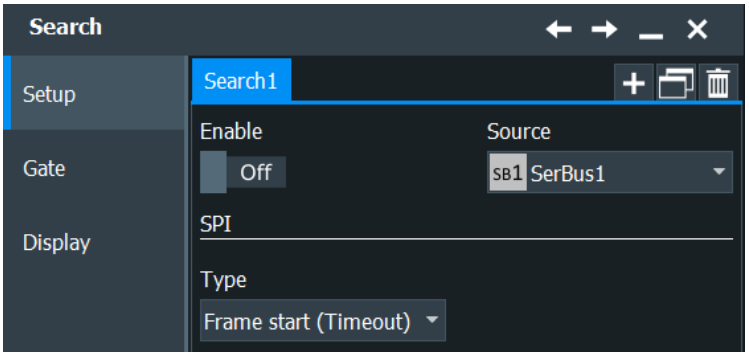
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 11, "Search functions", on page 419.

13.3.5.1 SPI search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"



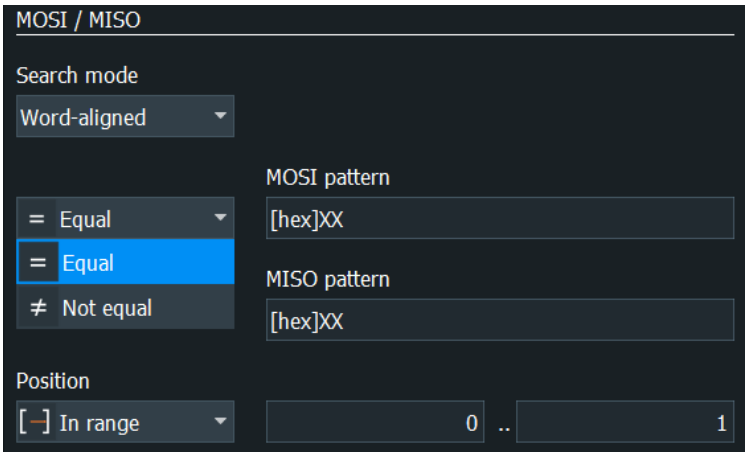
Type
Sets the event to be searched for.

"Frame start (SS)"	Searches for the start of the frame when slave select signal SS changes to the active state. This type is available if the slave select line is configured in the bus setup, and the frame condition is "SS".
"Frame start (Timeout)"	Searches for the start of the frame when the clock idle time exceeds the timeout. This trigger type is available if frame condition is set to "CLK timeout".
"MOSI"	Searches for a specified data pattern expected on the MOSI line.
"MISO"	Searches for a specified data pattern expected on the MISO line.
"MOSI / MISO"	Searches for specified data patterns expected on the MOSI and MISO lines.

Remote command:
`SEARch:TRIGger:SPI:TYPE` on page 1721

MOSI and MISO data search
The MOSI and MISO setup consists of the condition, position, MOSI pattern, MISO pattern, search mode (word-aligned, bit-aligned) and one or two index patterns.

The MOSI and MISO setup settings are the same as in the SPI trigger setup. For details, see "MOSI and MISO data conditions" on page 516.



Remote command:
`SEARch:TRIGger:SPI:FCONdition` on page 1721
`SEARch:TRIGger:SPI:MISopattern` on page 1722
`SEARch:TRIGger:SPI:MOSipattern` on page 1722
`SEARch:TRIGger:SPI:DPOPerator` on page 1722
`SEARch:TRIGger:SPI:DPOSition` on page 1722
`SEARch:TRIGger:SPI:DPTO` on page 1723
`SEARch:TRIGger:SPI:PALignment` on page 1723

13.3.5.2 SPI search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands are listed in Chapter 24.17.4.5, "SPI search results", on page 1723.

13.4 UART/RS-232/RS-422/RS-485 (option R&S RTO6-K510)

13.4.1 The UART / RS232 interface

The Universal Asynchronous Receiver/Transmitter UART converts a word of data into serial data, and vice versa. It is the base of many serial protocols such as RS-232. The UART uses only one line, or two lines for transmitter and receiver.

Data transfer

The data is transmitted in words, also referred to as symbols or characters. Each word consists of a start bit, several data bits, an optional parity bit, and one or more stop bits. Several words can form a package, or frame. The end of a package is marked with a reserved word or by a pause between two words.

Start	Data0	Data1	Data2	Data3	Data4	[Data5]	[Data6]	[Data7]	[Data8]	[Parity]	Stop

Figure 13-5: Bit order in a UART word (symbol)

- The start bit is a logic 0.
- The stop bits and the idle state are always logic 1.

The UART protocol has no clock for synchronization. The receiver synchronizes by means of the start and stop bits, and the bit rate that must be known to the receiver.

Trigger

The R&S RTO6 can trigger on specified parts of UART serial signals:

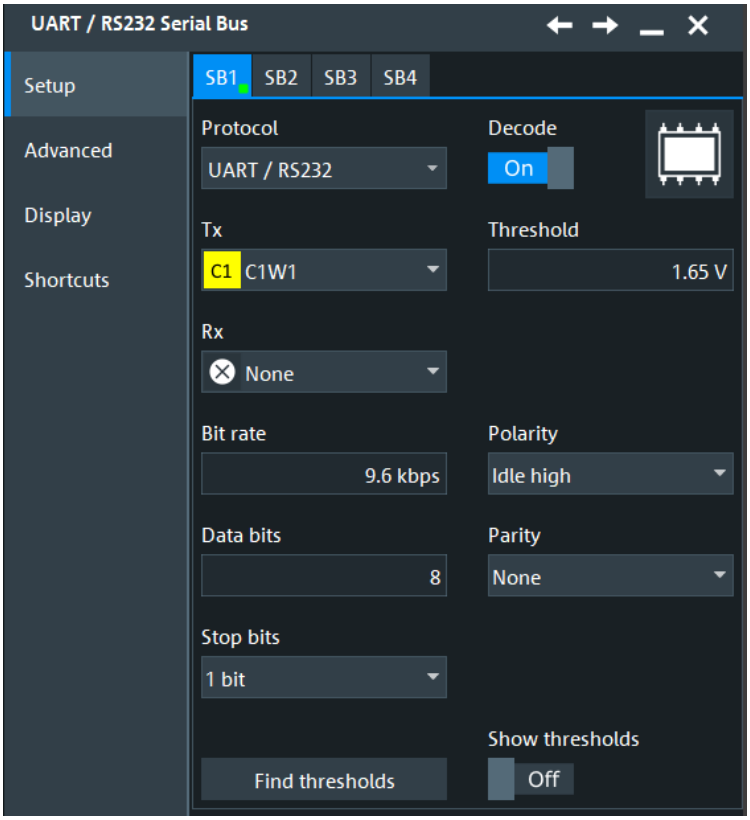
- Start bit
- Packet start
- Parity errors, and breaks
- Stop errors
- A serial pattern at any or a specified position

13.4.2 UART configuration

13.4.2.1 UART configuration settings

Access: [Bus] > "UART/RS232" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Source: Tx, Rx
Select the input channels for the transmitter and receiver signals.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Do not combine a reference waveform with channel or math waveform because the time correlation of these waveforms might differ.

Alternatively, digital channels can be used if MSO option R&S RTO6-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Remote command:
`BUS<m>:UART:TX:SOURce` on page 1728
`BUS<m>:UART:RX:SOURce` on page 1727

Threshold
Sets the threshold value for digitization of signals for each line. If the signal value on the line is higher than the threshold, the signal state is high. Otherwise, the signal state is considered low if the signal value is below the threshold. The interpretation of HIGH and LOW is defined by the Polarity.

Note: If the sources are digital channels, the same threshold values are used for the parallel and the serial buses. You can set the thresholds either in the parallel bus configuration or in the serial bus configuration.

"Threshold"	Enter the value directly in the field.
"Find thresholds"	Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
`BUS<m>:UART:RX:THReshold` on page 1728
`BUS<m>:UART:TX:THReshold` on page 1728
`BUS<m>:UART:TECHnology` on page 1729
`BUS<m>:SETReflevels` on page 1671
`BUS<m>:FAUToset` on page 1671

Polarity
Defines the logic levels of the bus. The idle state corresponds to a logic 1. the start bit to a logic 0. "Idle high" (high=1) is used, for example, for control signals, while "Idle low" (low=1) is defined for data lines (RS-232).

Remote command:
`BUS<m>:UART:POLarity` on page 1730

Bit rate
Sets the number of transmitted bits per second. To select a bit rate from list of predefined values, tap the icon beside the "Bit rate" field. To enter a specific value, open the keypad. The list of predefined values is also available in the keypad.

Remote command:
`BUS<m>:UART:BITRate` on page 1729
`BUS<m>:UART:BAUDrate` on page 1730

Data bits
Sets the number of data bits of a word in a range from 5 bits to 8 bits. If no parity bit is used, then 9 data bits are possible.

Remote command:
`BUS<m>:UART:SSIZe` on page 1731

Stop bits
Sets the number of stop bits: 1 or 1.5 or 2 stop bits are possible.

Remote command:
`BUS<m>:UART:SBIT` on page 1731

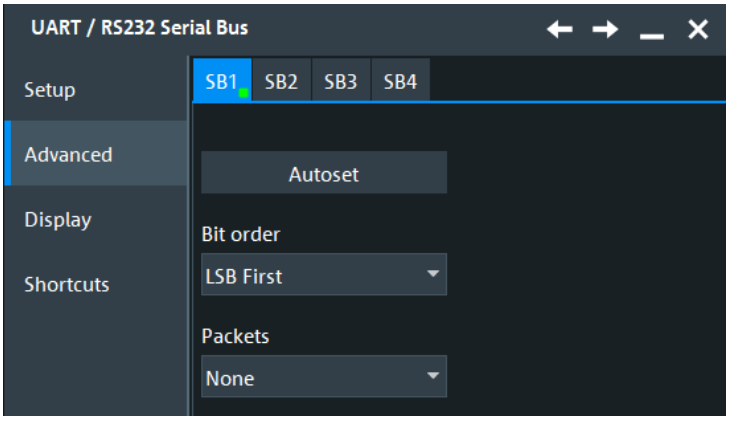
Parity
Defines the optional parity bit that is used for error detection.

"None"	No parity bit is used.
"Odd"	The parity bit is set to "1" if the number of data bits set to "1" is even.
"Even"	The parity bit is set to "1" if the number of data bits set to "1" is odd.
"Mark"	The parity bit is always a logic 1.
"Space"	The parity bit is always a logic 0.
"Don't care"	The parity is ignored.

Remote command:
`BUS<m>:UART:PARity` on page 1730

13.4.2.2 Advanced settings

Access: [Bus] > "Advanced" tab.



Autoset
Starts software algorithms for determining the signal threshold levels and bitrate.

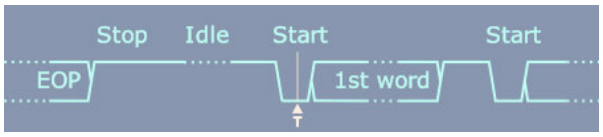
Remote command:
`BUS<m>:FAUToset` on page 1671

Bit order
Defines if a word starts with msb (most significant bit) or lsb (least significant bit). The display of the decoded signal considers this setting, results are displayed in the specified order.

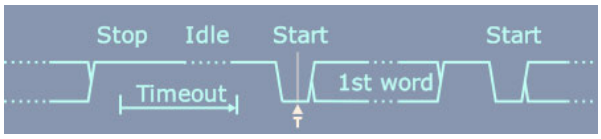
Packets
Allows you to define packets of several words in the data stream.

"None" Packets are not considered.

"End word" Defines a pattern as end condition of a packet, for example, a reserved word like CR or LF. The bit pattern editor provides frequently used values in the "Predefined values" list below the pattern table.
A new packet starts with the first start bit after the defined end pattern.



"Timeout" Defines a timeout between a stop bit and the next start bit. Enter the minimum time that marks the end of a packet.
A new packet starts with the first start bit after the timeout.



Remote command:
`BUS<m>:UART:PACKets` on page 1731
`BUS<m>:UART:TOUT` on page 1732
`BUS<m>:UART:EWORd` on page 1732

13.4.2.3 Configuring UART protocol

For details on configuration settings, see Chapter 13.4.2.1, "UART configuration settings", on page 521.

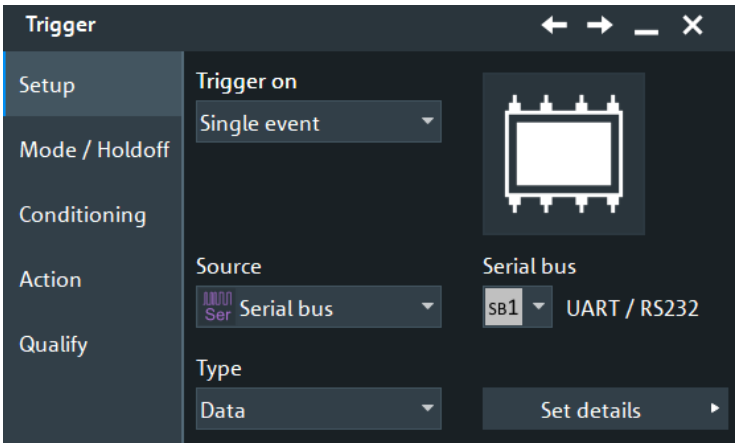
1. Press the [Bus] key on the front panel.
2. At the left-hand side, select the vertical tab of the bus you want to set up.
3. Select the "Setup" tab.
4. Tap the "Protocol" button and select the protocol: "UART".
5. Tap the "TX" button, and select the waveform of the transmitter line.
6. Tap the "Rx" button, and select the waveform of the receiver line.

7. Set the logical thresholds.
8. Set the "Bit rate" and "Stop bits".
9. Set the "Data bits" and "Parity".
10. Set the "Bit order", "Polarity" and "Packets".
11. Enable "Decode".

13.4.3 UART trigger

13.4.3.1 UART trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



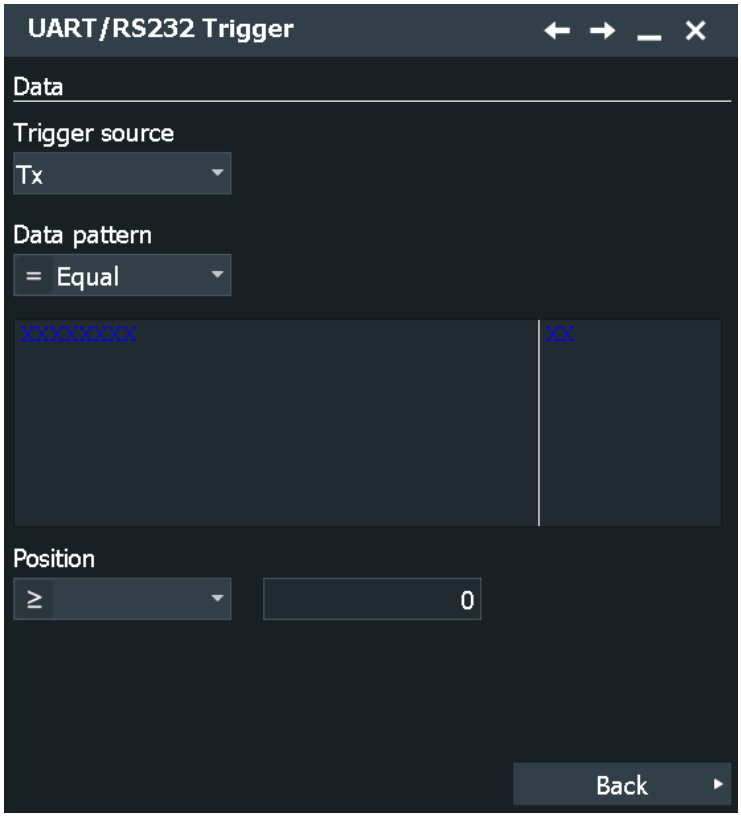
Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Type
Selects the trigger type for UART analysis.

"Start bit"	Triggers on a start bit. The start bit is the first low bit after a stop bit.
"Packet start"	Triggers on the begin of a data packet. This trigger type is available, if Packets is enabled.

"Data" Trigger on a serial pattern at a defined position in the data packet. The pattern can include several subsequent symbols (data frames).



See "Data conditions" on page 526.

"Parity error" Triggers on a parity error indicating a transmission error. This trigger type is only available if a parity is configured for the UART bus.

"Break condition" Triggers if a start bit is not followed by a stop bit, the data line remains at logic 0 for longer than a UART word.

"Stop error" Triggers if the stop bit is a logic 0.

Remote command:
`TRIGger<m>:UART:TYPE` on page 1733

Trigger source
Selects the transmitter or receiver line as trigger source.

Remote command:
`TRIGger<m>:UART:SOURce` on page 1733

Data conditions
Specify the data conditions if the trigger type is set to "Data".

Condition ← Data conditions
Selects the operator for the "Data" pattern: "Equal" or "Not equal".

Remote command:
`TRIGger<m>:UART:FCONdition` on page 1734

Pattern ← Data conditions
Specifies the data pattern to be found on the specified trigger source, in binary or hex format. Enter the words in msb first bit order. The starting point of the pattern is defined by Position and Index min, Index max.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

Remote command:
`TRIGger<m>:UART:DATA` on page 1734

Position ← Data conditions
Operator for the data position. You can define an exact position, or a position range.

The setting is available if Packets is enabled in the protocol configuration.

Remote command:
`TRIGger<m>:UART:DPOPerator` on page 1733

Index min, Index max ← Data conditions
Sets the number of words before the first word of interest. These offset words are ignored. If the Position defines a range, the first and the last words of interest are defined.

The setting is available if packet detection is enabled in the protocol configuration.

Remote command:
`TRIGger<m>:UART:DPOSition` on page 1734
`TRIGger<m>:UART:DPTO` on page 1734

13.4.3.2 Triggering on UART signals

Prerequisites: An UART bus is configured, see Chapter 13.4.3.1, "UART trigger settings", on page 525.

1. Press the [Bus] key and select the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to UART.
5. Select the "Trigger type" and "Trigger source".
6. For more complex trigger types, enter the data conditions: pattern, position, index. For details, see Chapter 13.2.3, "I²C trigger", on page 499

13.4.4 UART decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Serial Bus" dialog > "Setup" tab, enable "Decode".

2. In the "Serial Bus" dialog > "Display" tab, select additional result display settings: "Show table" and "Show binary".
3. If required, enable "Zoom coupling"

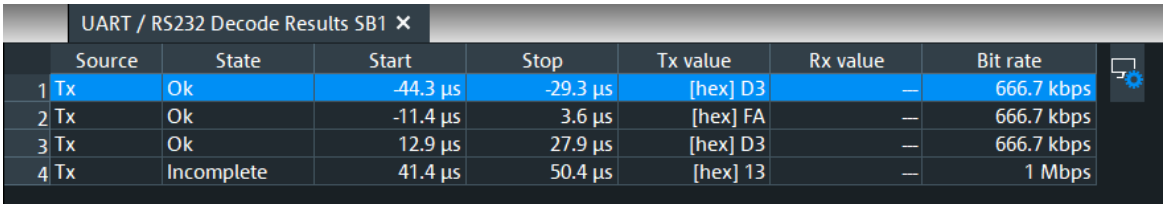
For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The "Decode results" box shows the detailed decoded data for each word.



UART / RS232 Decode Results SB1

	Source	State	Start	Stop	Tx value	Rx value	Bit rate
1	Tx	Ok	-44.3 µs	-29.3 µs	[hex] D3	—	666.7 kbps
2	Tx	Ok	-11.4 µs	3.6 µs	[hex] FA	—	666.7 kbps
3	Tx	Ok	12.9 µs	27.9 µs	[hex] D3	—	666.7 kbps
4	Tx	Incomplete	41.4 µs	50.4 µs	[hex] 13	—	1 Mbps

The decoding process considers the "Bit order" configuration setting of the signal and displays the binary result MSB first.

Table 13-5: Content of the Decode results table

Column	Description
Number of Frames	
Source	Line, Tx or Rx
State	Decoding state of the word. "Insuffcient waveform length" indicates that the word is not completely contained in the acquisition. Change the horizontal scale, or move the reference point to the left to get a longer acquisition.
Start	Time of the word start (start bit)
Stop	Time of the word stop (stop bit)
Tx value	Value of the Tx word. The data format is selected in the "Display" tab.
Rx value	Value of the Rx word. The data format is selected in the "Display" tab.
Bit rate	Value of the bit rate .

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands to retrieve decode results are described in Chapter 24.17.5.3, "Decode results", on page 1735.

13.5 Audio signals (option R&S RTO6-K510)

The R&S RTO6 can analyze several standard and de-facto industry standard signals: I²S Inter-IC Sound standard audio format, left justified and right justified data formats and Time Division Multiplexed (TDM) audio format.

- Audio protocols 529
- Audio configuration 531
- Audio trigger 537
- Audio decode results 541
- Track 542
- Trend 545

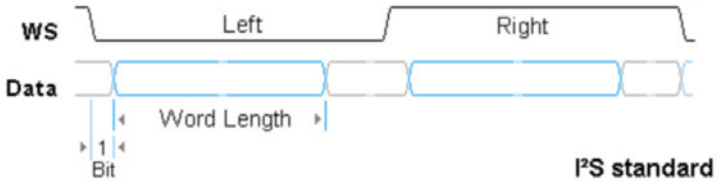
13.5.1 Audio protocols

All audio protocols use 3 lines:

- The clock line generates the bit clock.
- The word select line (WS, also known as word clock) defines the frame start and the maximum length of the data word.
  For pulse code modulated signals (I²S standard, left and right justified data formats), the level of the WS signal assigns the data words to the left and right channels.
  TDM uses frame synchronization pulses on the WS line to identify the beginning of a frame.
- The data line transmits the audio data in time-multiplexed data channels.

13.5.1.1 I²S standard

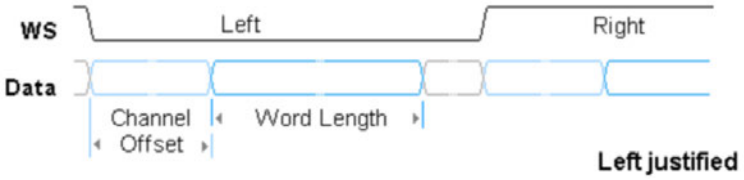
I²S standard interfaces transmit two PCM coded audio channels. The WS line selects the channel being transmitted - left or right channel. Usually, 32 bits are transmitted on each channel. The data word can be shorter than the channel length, and the receiver ignores the remaining bits. The first byte of the audio word is delayed one clock period from the leading edge of the word select pulse. The R&S RTO6 can decode I²S standard signals with MSBF and LSBF bit order.



13.5.1.2 Left justified data format

The left justified data format is very similar to the I²S standard, but the first byte of the audio word is aligned with the leading edge of the word select pulse. Thus the audio word is left justified within the frame. The data word can be shorter than the channel length.

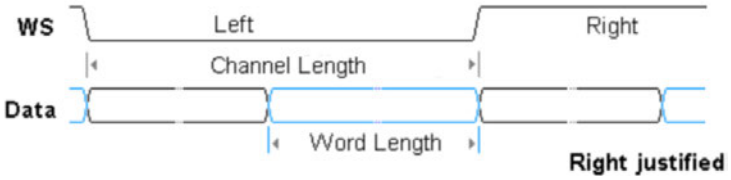
In addition to the standard configuration, the R&S RTO6 can analyze also left justified data formats which send the data word with offset to the WS edge. The bit order can be MSBF or LSBF.



13.5.1.3 Right justified data format

The right-justified data format is similar to the left-justified, but the last byte of the word in the frame is aligned with the trailing edge of the word select pulse. Thus the audio word is right-aligned within the frame.

The R&S RTO6 can decode right justified signals with MSBF and LSBF bit order.



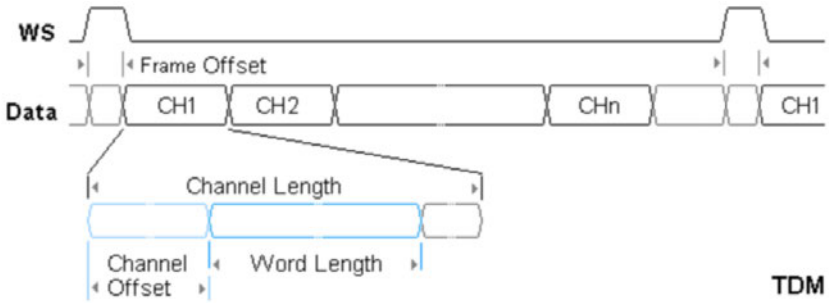
13.5.1.4 TDM

The Time Division Multiplexed (TDM) audio format is not standardized and provides high flexibility for transfer of more than two audio data channels on one line. On the word select line, it uses frame synchronization pulses to identify the beginning of a frame. On the data line, channel blocks of a defined length are transmitted. Each block contains an audio word that can be shorter than the channel length.

Each frame can start with frame offset bits, which preceed the first channel. Inside the channel, the audio word also can have an offset to the channel start.

Channel length, channel offset and word length are dependent values:

Channel length ≥ Word length + Channel offset



13.5.2 Audio configuration

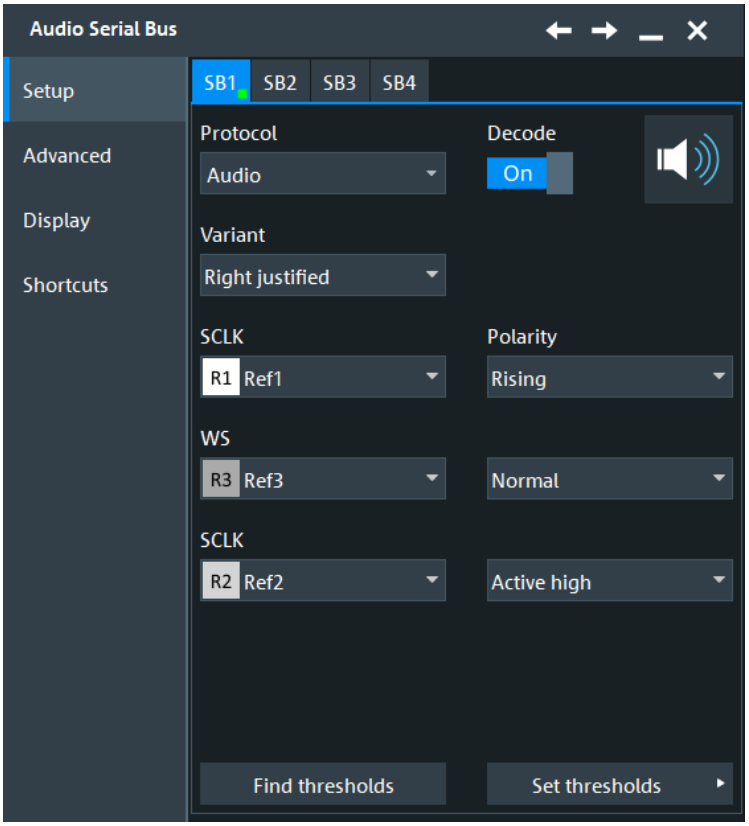
13.5.2.1 Audio signal configuration

Access: [Bus] > "Audio" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.

In the "Setup" tab, you configure the audio signal. Several audio signal variants are available: the I²S standard signal, the left- and right-justified data formats, and the TDM interface.

For all audio signal variants, you define the line sources and their polarities. Also, if coupling is active, one threshold for all sources; if coupling is not enabled, three thresholds for each source.



Audio Variant

Selects the protocol variant of the audio signal. The configuration possibilities exceed the definitions of the standards.

"I2S Standard" Inter-IC sound standard audio format.
It uses the SCLK, WS and SDATA lines. The first byte of the audio word is delayed one clock period from the leading edge of the word select pulse.

"Left justified" The left-justified data format uses the same lines as I²S standard. The first byte of the audio word is aligned with the leading edge of the word select pulse, or left justified within the frame. The format is word-length independent.

"Right justified" The right-justified data format is similar to the left-justified, but the last byte of the last word in the frame is aligned with the trailing edge of the word select pulse, or right-aligned within the frame. This format is not word-length independent.

"TDM" The Time Division Multiplexed audio format is not standardized and provides high flexibility for transfer of up to 8 audio data channels on one line. Instead of words select, it uses frame synchronization pulses to identify the beginning of a frame. On the data line, channel blocks of a defined length are transmitted. Each block contains an audio word followed by several zero bits to complete the block.

Remote command:
`BUS<m>:I2S:AVARiant` on page 1737

SCLK
Selects the source of the clock line.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Do not combine a reference waveform with channel or math waveform because the time correlation of these waveforms might differ.

Alternatively, digital channels can be used if MSO option R&S RTO6-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Remote command:
`BUS<m>:I2S:CLOCk:SOURce` on page 1738

SCLK Polarity
Sets th polarity of the clock signal, that is the edge at which the instrument samples the data on the data line. Usually, the rising edge is used. The R&S RTO6 can also analyze the converse setup.

Remote command:
`BUS<m>:I2S:CLOCk:POLarity` on page 1738

WS / FSYNC
Selects the source of the word select line for I²S standard, left- and right-justified data formats, or the source of the frame synchronization pulse for TDM audio signals. The same waveforms as for SCLK are available, and the same restrictions are applied.

Remote command:
`BUS<m>:I2S:WSELect:SOURce` on page 1738

WS / FSYNC Polarity
For a word select line, the polarity defines the word select values assigned to the left and right channels.
- "Normal": usually, 0 indicates the left channel, and 1 indicates the right channel.
- "Inverted": 0 indicates the right channel, and 1 the left channel.

For an FSYNC line (TDM), the polarity defines the edge of the FSYNC pulse that identifies the beginning of a frame. The frame starts exactly at the next clock edge following the selected FSYNC edge.
- "Normal": usually, the frame begins with a rising edge.
- "Inverted": the frame begins with a falling edge.

Remote command:
`BUS<m>:I2S:WSELect:POLarity` on page 1739

SDATA
Selects the source of the audio data line. The same waveforms as for SCLK are available, and the same restrictions are applied.

Remote command:
`BUS<m>:I2S:DATA:SOURce` on page 1739

SDATA Polarity

Defines the interpretation of high and low signal states.

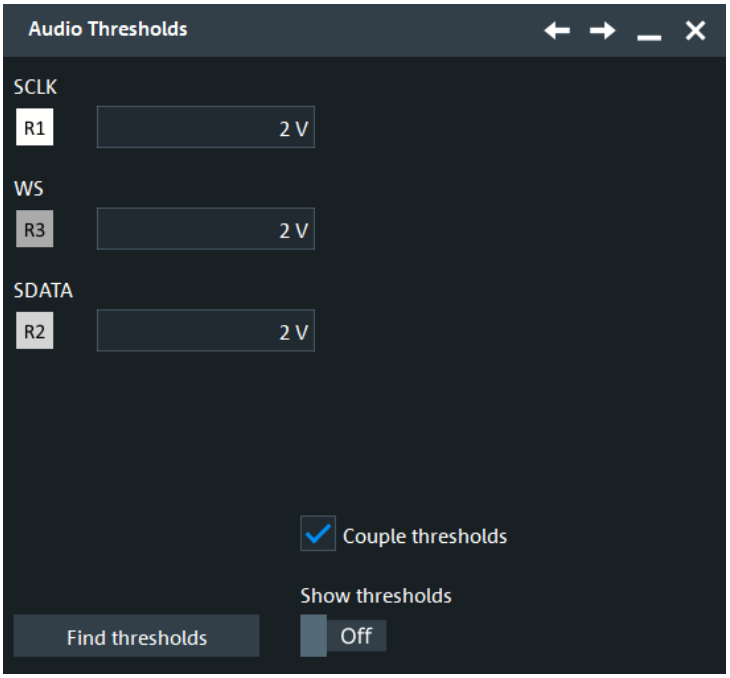
- "Active high": HIGH (signal level above the threshold level) = 1 and LOW (signal level below the threshold level) = 0
- "Active low": HIGH = 0 and LOW = 1

Remote command:

`BUS<m>:I2S:DATA:POLarity` on page 1740

Threshold setup

Opens a dialog to set the thresholds value for digitization of signals for each line.



If the signal value on the line is higher than the threshold, the signal state is high. Otherwise, the signal state is considered low if the signal value is below the threshold. The interpretation of HIGH and LOW is defined by the polarity.

There are three ways to set the threshold:

"Threshold"	Enter individual values for each line directly in the fields.
"Find thresholds"	Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude. This option is only available for analog sources.
"Couple thresholds"	Sets all thresholds to the same value. Enter the value in the "Threshold" field.

Remote command:

`BUS<m>:I2S:TCOupling` on page 1740
`BUS<m>:I2S:CLOCk:THReshold` on page 1740
`BUS<m>:I2S:DATA:THReshold` on page 1741
`BUS<m>:I2S:WSELect:THReshold` on page 1740
`BUS<m>:SETReflevels` on page 1671

13.5.2.2 Advanced settings

Access: [Bus] > "Advanced" tab.

The settings available in the "Advanced" dialog depend on the selected Audio Variant.

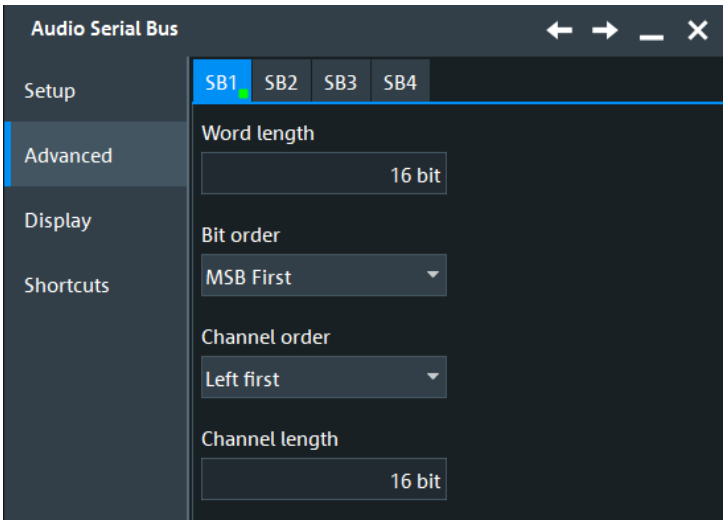


Figure 13-6: Advanced Audio settings for Variant = Right justified

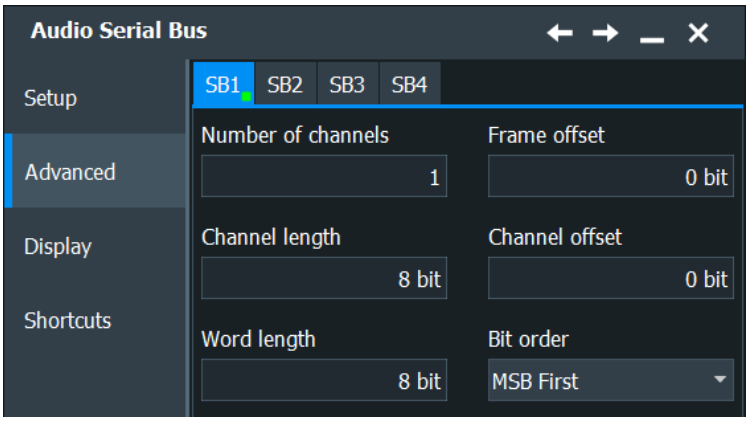


Figure 13-7: Advanced Audio settings for Variant = TDM

Word length
Defines the number of bits in an audio data word. The minimum length is 4 bit, the maximum is 32 bit.

Remote command:
`BUS<m>:I2S:WLENgth` on page 1741

Bit order
Sets the bit order in the audio data words. Usually, the MSB is transmitted first.

Remote command:
`BUS<m>:I2S:BORDer` on page 1742

Channel order
Defines if the left or the right channel is the first channel in the frame.

The setting is not available for TDM audio signals.

Remote command:
`BUS<m>:I2S:CHANnel:ORDer` on page 1741

Channel offset
Available for "Variant" = "Left justified"/ "TDM".

Sets the number of bits between the channel start and the start of the audio word. The setting is available for left-justified data format and TDM audio signals.

For TDM, possible values depend on the channel size and the word size. The maximum delay is *Channel length - Word length*.

Remote command:
`BUS<m>:I2S:CHANnel:OFFSet` on page 1742

Number of channels
Available for "Variant" = "TDM".

Sets the number of channels transmitted on the TDM audio line.

Remote command:
`BUS<m>:I2S:CHANnel:TDMCount` on page 1742

Frame offset
Available for "Variant" = "TDM".

Sets the number of bits between the frame start and the start of the first channel of a TDM audio line. The maximum offset is 256 bit. Each FSYNC edge restarts the offset count.

Remote command:
`BUS<m>:I2S:FOFFset` on page 1742

Channel length
Available for "Variant" = "Left justified" / "Right justified" / "TDM"

Sets the number of bits a frame contains.

Remote command:
`BUS<m>:I2S:CHANnel:LENGth` on page 1743

13.5.2.3 Configuring audio protocol

The configuration of the Audio is simple - assign the two lines to input channels, and set the thresholds.

For details on configuration settings, see Chapter 13.5.2.1, "Audio signal configuration", on page 531.

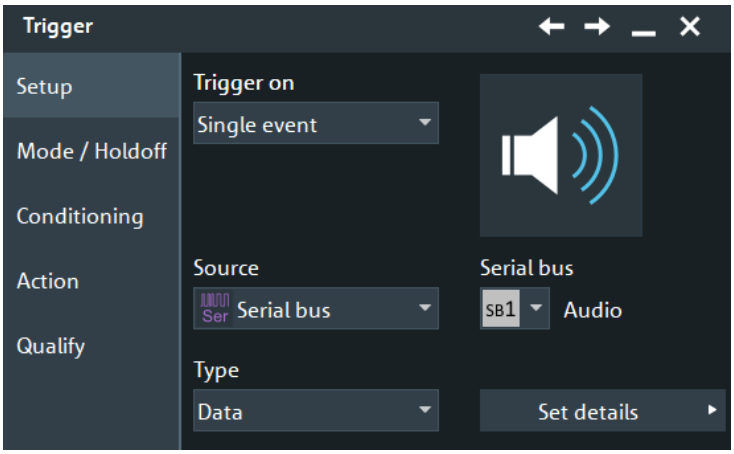
1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".

3. Select the "Setup" tab.
4. Tap the "Protocol" button and select the protocol: "Audio".
5. Select the "Audio Variant".
6. Set the parameters, as required according to the selected "Audio Variant".
7. Enable "Decode", if available.

13.5.3 Audio trigger

13.5.3.1 Audio trigger

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Serial bus
Selects the serial bus to be triggered on. Make sure to select the correct bus before you enter the settings.

To trigger on a serial bus, the signals sources must be channel signals. If the data or clock source is a math or reference waveform, you cannot trigger on that bus.

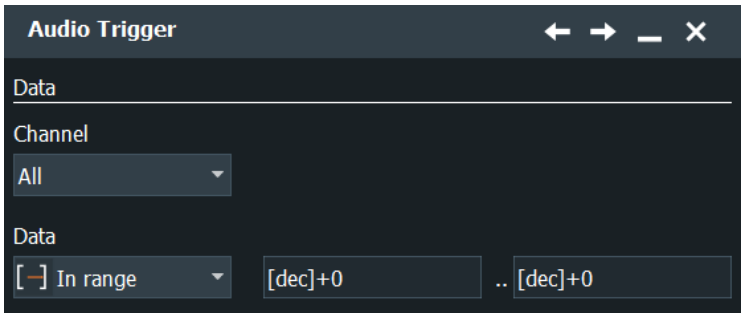
Remote command:
`TRIGger<m>:SOURce:SBSelect` on page 1675

Protocol
Defines the protocol type of the selected serial bus.

Remote command:
`BUS<m>:TYPE` on page 1670

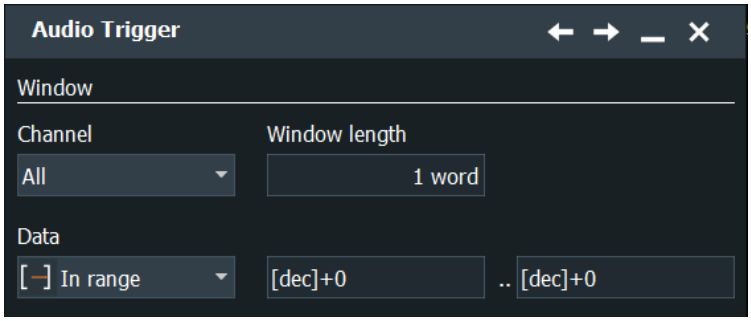
Type
Selects the trigger type.

"Data" Sets the trigger on a data word or a data range that occurs on a specified channel or on any channel. The instrument triggers on the last bit of the specified data pattern.



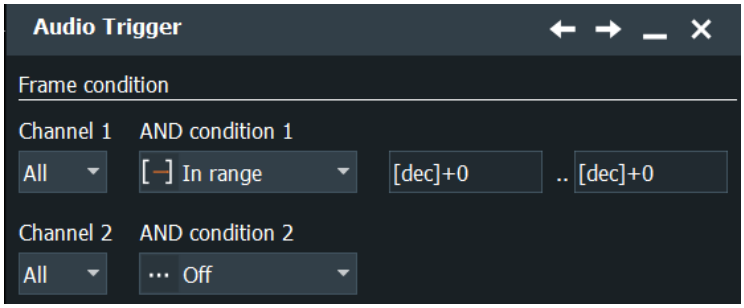
Description of specific trigger type settings:
- "Channel" on page 539
- "Data condition" on page 539

"Window" This trigger checks if the decoded data values stay inside a "window" that is formed by a data range and a time specified by a number of subsequent words. It considers a selected channel or all channels. The instrument triggers at the end of the last word. Thus, for example, you can trigger on a pause.



Description of specific trigger type settings:
- "Channel" on page 539
- "Data condition" on page 539
- " Window length" on page 540

"Frame condition" Sets the trigger on an AND combination of data conditions on different channels. The instrument triggers if all conditions are met inside one frame.



Description of specific trigger type settings:
- "Channel" on page 539
- "Data condition" on page 539

Description of specific trigger type settings: "Channel" on page 539 and .

"Word select" Triggers on the selected edge of the WS line, that is, on the beginning of the left or right channel (I²S, left- and right-justified). For TDM signals, it triggers on the selected edge of the FSYNC line - on the beginning of a TDM frame.
The trigger time is the first clock edge after the selected WS/FSYNC edge.
Description of specific trigger type settings: "Word select: Slope" on page 540.

"Error condition" The oscilloscope uses the WS or FSYNC line to monitor the channel and frame length. An error is detected when two consecutive frames have different length. The instrument triggers on the first clock edge after error detection.

Remote command:
`TRIGger<m>:I2S:TYPE` on page 1743

Channel
Selects the audio channel on which the instrument looks for the specified data condition.

The setting is relevant for trigger types Data, Window and Frame condition.

Note: For TDM signals. the number of available channels depends on the configuration of the audio bus, see Chapter 13.5.2.2, "Advanced settings", on page 535.

Remote command:
`TRIGger<m>:I2S:TCONdition<n>:CHANnel` on page 1744

Data condition
The data condition setup consists of the operator and one or two data patterns.

The settings are relevant for trigger types Data, Window and Frame condition.

"Operator" Defines the operator to set a specific data word ("Equal" or "Not equal") or a data range.

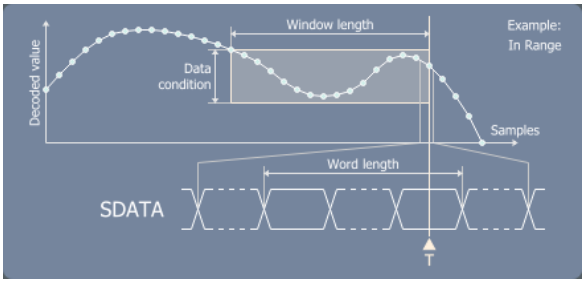
"Min data" Defines the data pattern. The data length is limited to the word length. Enter the pattern using the bit order defined in the signal configuration. X (don't care) is not allowed.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Max data" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:I2S:TCONdition<n>:CONDtion` on page 1745
`TRIGger<m>:I2S:TCONdition<n>:DMIN` on page 1746
`TRIGger<m>:I2S:TCONdition<n>:DMAX` on page 1746

Window length
Sets the number of words that is used as time limit for the "Window" trigger type. The instrument triggers if the data condition is fulfilled on the same channel for the given number of subsequent frames.



Remote command:
`TRIGger<m>:I2S:SOWords` on page 1746

Word select: Slope
Sets the edge of the WS or FSYNC signal as trigger condition. The instrument triggers on the first clock edge after the specified edge.

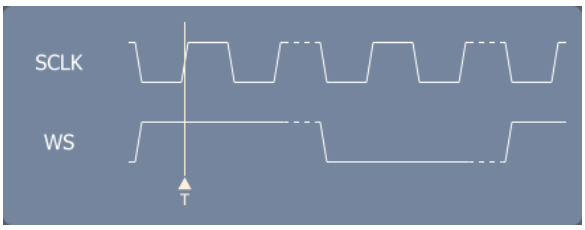


Figure 13-8: Word select trigger on I²S standard signal with clock polarity "Rising" and "Normal" WS polarity (left = 0)

The WS edge indicates the start of the left or right channel. The FSYNC edge indicates the frame start.

Consider the Audio signal configuration setting in the "Setup" dialog box.

Remote command:
`TRIGger<m>:I2S:WSSLope` on page 1747

13.5.3.2 Triggering on audio signals

Prerequisites: An Audio bus is configured, see Chapter 13.5.2, "Audio configuration", on page 531.

1. Press the [Bus] key and select the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to Audio.
5. Select the "Trigger type".
6. For more complex trigger types, enter the data, frame or word select conditions. For details, see Chapter 13.5.3.1, "Audio trigger", on page 537

13.5.4 Audio decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Serial Bus" dialog > "Setup" tab, enable "Decode".
2. In the "Serial Bus" dialog > "Display" tab, select additional result display settings: "Show table" and "Show binary".
3. If required, enable "Zoom coupling"

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

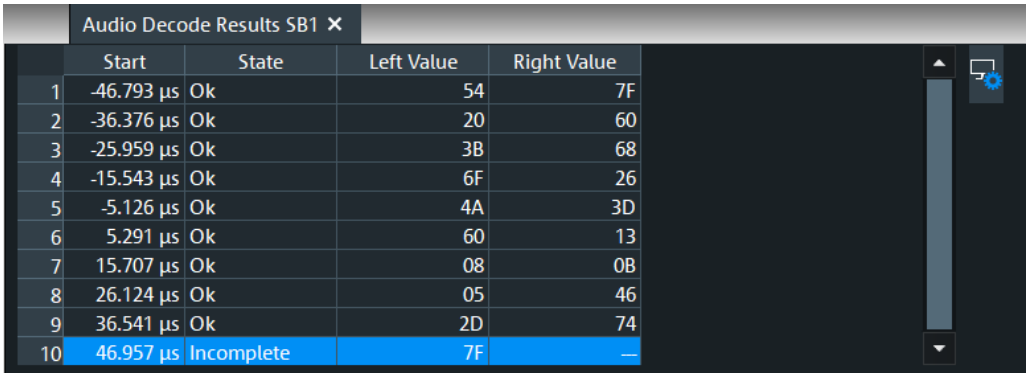
The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.



Audio Decode Results SB1

	Start	State	Left Value	Right Value
1	-46.793 µs	Ok	54	7F
2	-36.376 µs	Ok	20	60
3	-25.959 µs	Ok	3B	68
4	-15.543 µs	Ok	6F	26
5	-5.126 µs	Ok	4A	3D
6	5.291 µs	Ok	60	13
7	15.707 µs	Ok	08	0B
8	26.124 µs	Ok	05	46
9	36.541 µs	Ok	2D	74
10	46.957 µs	Incomplete	7F	—

The results are shown in the "Decode results" table.

Table 13-6: Content of the Decode results table

Column	Description
Start	Time of the frame start
State	Overall state of the frame. "Insuffcient waveform length" indicates that the frame is not completely contained in the acquisition.
Left Value	Data value of the left channel
Right Value	Data value of the right channel

Table 13-7: Content of the Decode results table

Column	Description
Frame Start	Time of the frame start
Frame State	Overall state of the frame. "Insuffcient waveform length" indicates that the frame is not completely contained in the acquisition.
TDM<x> Value	Data value of the TDM channel

Remote commands

Remote commands are described in Chapter 24.17.6.3, "Decode results", on page 1747.

13.5.5 Track

The track is a waveform that shows measurement values in time-correlation to the audio signal. It is the graphical interpretation of all measurement values of a single acquisition. For audio signals, the measurement values on the vertical axis are the decoded values of the audio channels, the time scale is equivalent to the scale of the source waveforms.

You can display the values of several channels in one track, or create one track for each channel and display them in parallel.

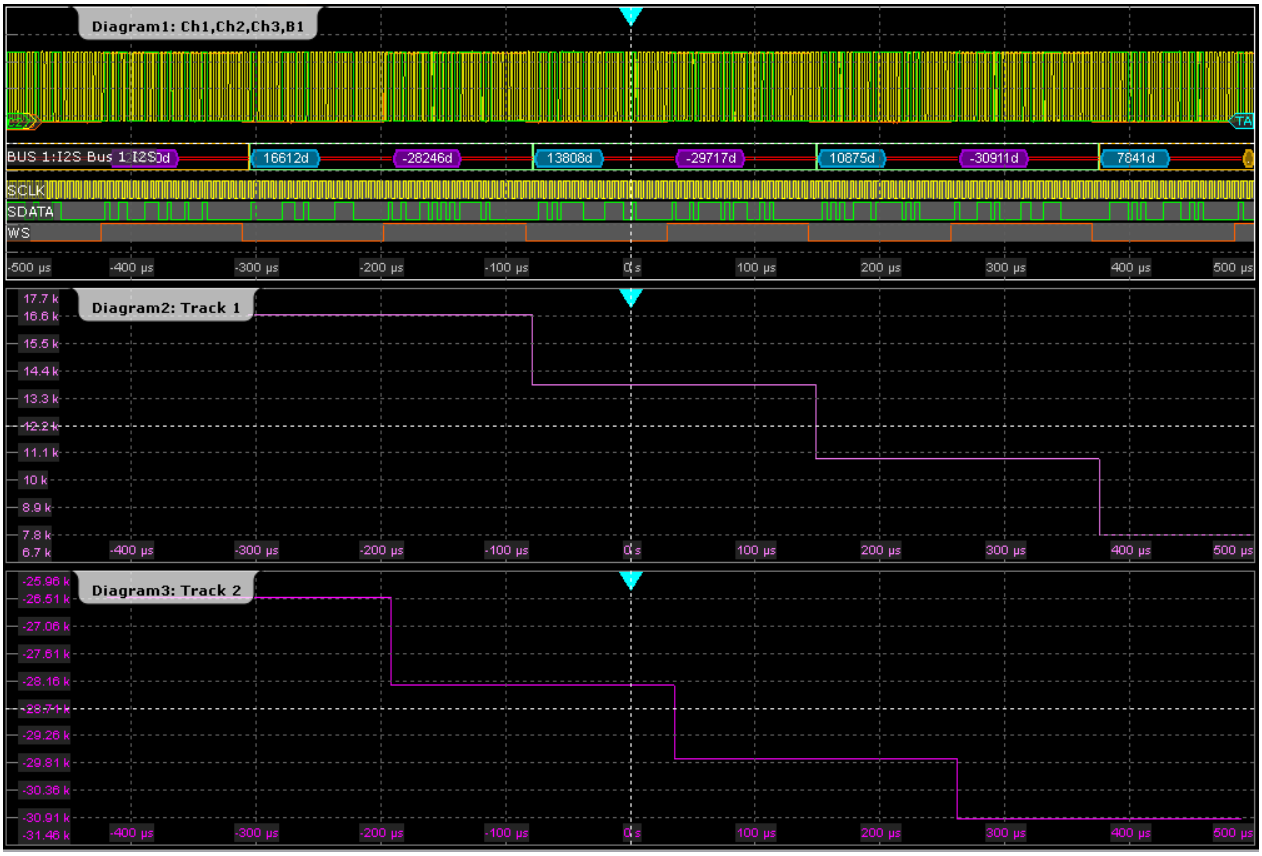


Figure 13-9: Tracks of left and right channel of I²S standard signal

The track is a special measurement waveform, so it can be used for further analysis like cursor measurements and zoom.

The instrument uses the bus data format to interpret audio data ("Display" tab > "Data format"). ASCII format is not supported.

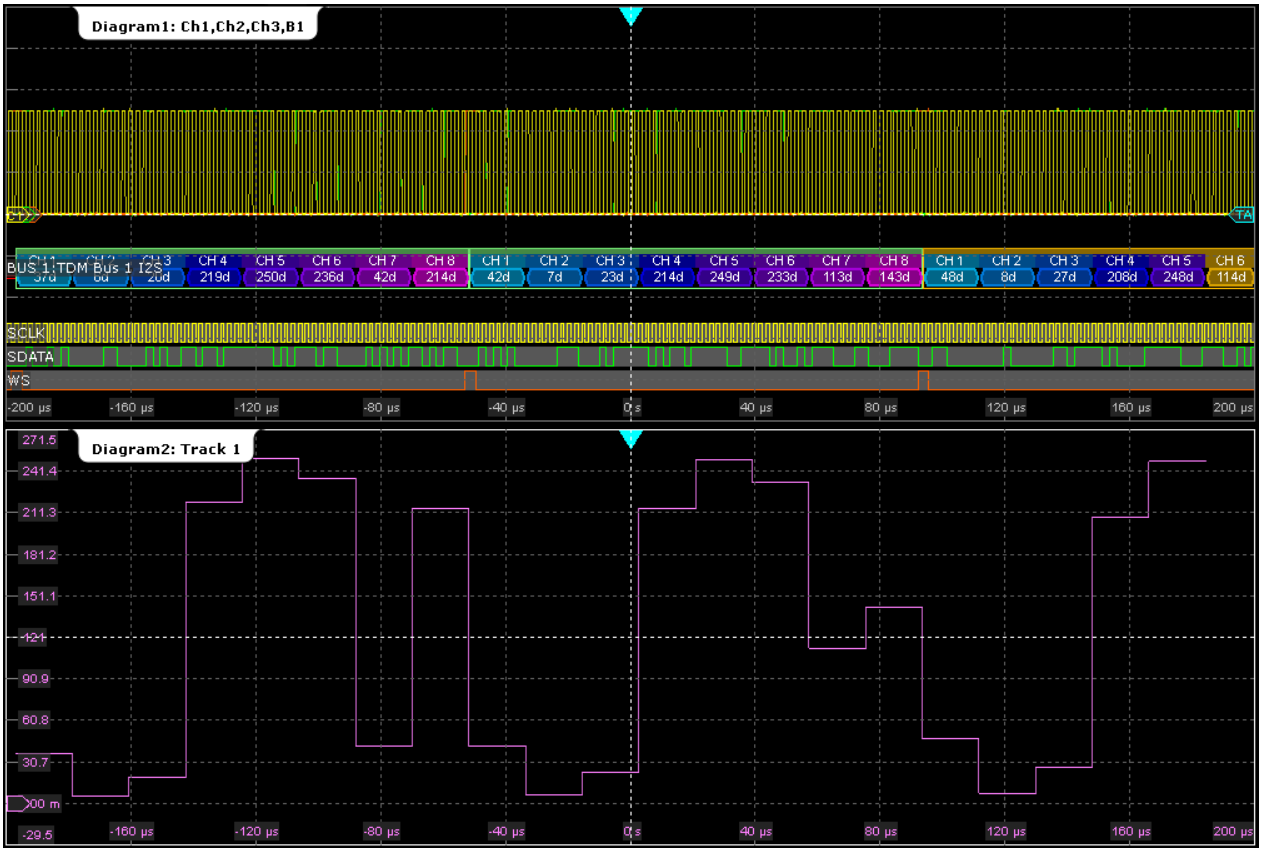
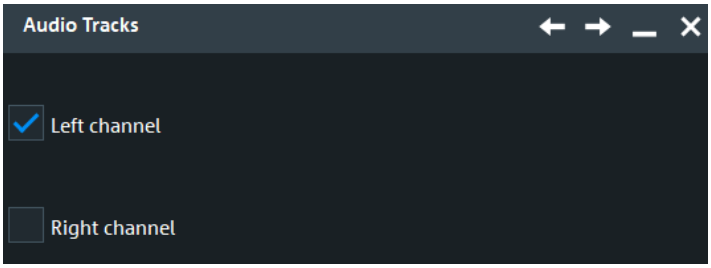


Figure 13-10: Track of all 8 channels of a TDM signal

13.5.5.1 Track settings in protocol setup

You can enable the track waveforms in the protocol settings.



Show Track Waveform(s)
Enables and displays the tracks for the selected channels of the decoded bus using continuous auto scale.

See also: Chapter 13.5.5, "Track", on page 542.

Remote command:
`BUS<m>:I2S:TRACk:LEFT` on page 1750
`BUS<m>:I2S:TRACk:RIGHt` on page 1750
`BUS<m>:I2S:TRACk:TD1Ch` on page 1750

BUS<m>:I2S:TRACk:TD2Ch on page 1750
BUS<m>:I2S:TRACk:TD3Ch on page 1750
BUS<m>:I2S:TRACk:TD4Ch on page 1750
BUS<m>:I2S:TRACk:TD5Ch on page 1751
BUS<m>:I2S:TRACk:TD6Ch on page 1751
BUS<m>:I2S:TRACk:TD7Ch on page 1751
BUS<m>:I2S:TRACk:TD8Ch on page 1751

13.5.5.2 Track enabling in measurement setup

As the track is based on measurement, it can be set up in the "Result Analysis" tab of the "Measurement" dialog box. Alternatively, you can enable the track in the protocol display settings, see Chapter 13.5.5.1, "Track settings in protocol setup", on page 544.

To set the vertical scale of the track waveform, use analysis scale settings, see "Analysis scale" on page 361.

Enable (Track)
Enables the track of measurement results over time and displays the track waveform. It is the graphical interpretation of all measurement values of a single acquisition.

The track is available for most amplitude/time measurements (except for High, Low, Amplitude, Max, Min, Peak to peak, Mean, RMS, S-dev, Pos. and Neg. overshoot, and Area), and for jitter measurements.

Enabling the track enables also the Continuous auto scale and Measure all events in each acquisition.

Before you can enable the track, activate the appropriate measurement.

Remote command:
MEASurement<m>:TRACk[:STATe] on page 1545

13.5.6 Trend

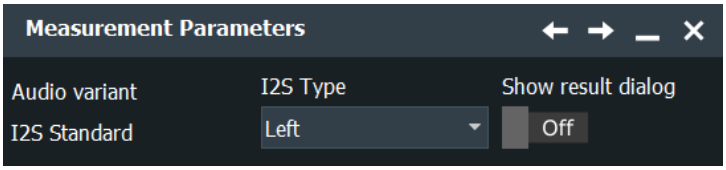
The trend is a special long-term measurement that shows the evolution of measurement values in a running continuous acquisitions. For audio signals, each decoded channel value is a measurement result that creates a point on the trend curve. You can configure the number of points that builds the complete trend curve.

13.5.6.1 Displaying and configuring the trend

If an audio bus is configured, you can set up the trend measurement for audio channels. The following procedure describes the complete trend setup using the "Setup" dialog.

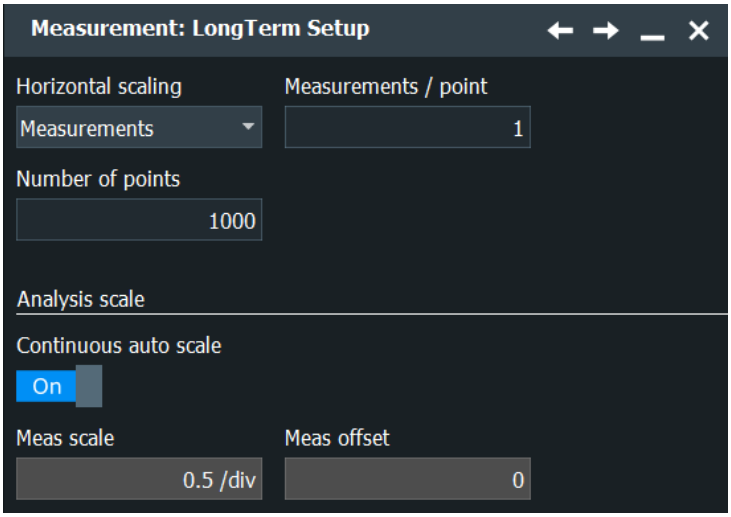
1. Select "Setup".
2. Select the "Category" = "Protocol".

3. Select the "Source" of the measurement: "Serial bus". Select the bus configured for the audio signal.
4. Enable the measurement.
5. Tap "Add / Remove " and select "Audio".
6. Select the "Audio channel" for which you want to analyze the trend.



7. In the "Measurements" dialog, select the "Plot" tab.
8. Under "Long term", select "Enable".

   An empty trend diagram is displayed.
9. Tap "Settings".
10. In the "Measurement: Long Term Setup", set the "Horizontal scaling" to "Measurements".



11. Set the number of "Measurements/point" to *1*.
12. Enable "Continuous auto scale".
13. Start continuous acquisition.

    The trend fills up with measurement points from left to right.

13.6 Custom: Manchester / NRZ (option R&S RTO6-K510)

R&S RTO6-K510 is a firmware option that enables the R&S RTO6 to analyze customizable serial bus signals encoded by the following coding standards:

- Manchester
- Manchester II
- NRZ Clocked
- NRZ Unclocked

For analysis, signals encoded in any of these protocols can be triggered and decoded.

Due to the free format description, no search within the decoded events is available.

This chapter describes:

- Custom: Manchester / NRZ protocols....547
- Custom: Manchester / NRZ configuration....549
- Custom: Manchester / NRZ trigger....567
- Custom filter....572
- Custom: Manchester / NRZ decode results....575
- Search on decoded custom Manchester / NRZ data....576

13.6.1 Custom: Manchester / NRZ protocols

"Manchester" coding is a self-clocked coding scheme also known as phase-shift keying (or phase encoding, PE). It is used in protocols such as ProfiBus (IEC 61158), DALI (Digital Addressable Lighting Interface, IEC 60929 and IEC 62386), MVB (Multifunction Vehicle Bus, part of IEC 61375 for Train Communication Networks, TCN), and Ethernet 10BASE-T (10 Mbit/s, IEEE 802.3i). In terms of a logical Boolean operation, the Manchester value of each bit (as per G. E. Thomas) is the exclusive disjunction (XOR) of the original data value and the clock value. A "0" is expressed by a high-to-low transition, a "1" by a low-to-high transition. These transitions, which occur at the middle of each bit period, make the signal self-clocked.

"Manchester II" coding (as per IEEE 802.3) is represented by inverted Manchester values: a "0" is expressed by a low-to-high transition, a "1" by a high-to-low transition.

NRZ stands for "non-return-to-zero" coding: Typically a "1" is represented by a positive voltage and a "0" is represented by a negative voltage, with no "zero" voltage state. NRZ code requires only half the bandwidth of Manchester code, and it can either be clocked or unclocked. NRZ unclocked signals require a user-defined bit rate and gap time setting for triggering and decoding.

13.6.1.1 Special features of Manchester coding

In practical protocols, Manchester coding appears in many variations, often employing deliberate coding violations to encode special waveform features, such as unambiguous synchronization and termination patterns. To adapt to these specific Manchester implementations and handle ambiguous signals, the option R&S RTO6-K510 for Cus-

tom Serial Bus uses a combination of automatic algorithms and user configurable parameters.

Quaternary Symbols

The software supports not just traditional binary symbols "0" and "1", but also arbitrary violation waveforms that use two additional symbols, yielding a total of four valid "quaternary bit" values. The two additional violation symbols are "H" (high) and "L" (low). Values of "H" correspond to a waveform lacking a transition in the center of the bit, with a physical high voltage state. Similarly, "L" violations also lack a center transition, but have a physical low voltage state. Most Manchester synchronization and termination conventions, even those containing violations, may be expressed as sequences of these four symbols. R&S RTO6-K510 uses the quaternary notation to support Manchester patterns in the honeycomb display and to describe synchronization and termination patterns in the frame description table.

Idle Conditions

The state of the signal line in between messages is the idle condition. Manchester appears in practical standards with varying idle conditions: it can idle at the high, low, or middle voltage state. High and low idle states correspond to "biphase" Manchester, while the middle voltage (often ground) adds a third state to become "ternary" Manchester. Using ternary Manchester, option R&S RTO6-K510 can usually establish the gaps between messages automatically. Using binary Manchester, the software has no way to automatically discriminate an idling bus from monotonic sequences of "H" or "L" violations. For these biphase situations, R&S RTO6-K510 offers a "Gap Time" detection feature, which allows to distinguish long intervals of non-transitions between bus idling and sequences of violations. Other differences between biphase and ternary Manchester are managed automatically by the software, with no user input required.

Edge Conventions

Most Manchester encodings establish the beginning of the first bit by a first transition, hence an "overhead" edge. The center of the bit is then marked by a second transition, which is a "sampling" edge. Some Manchester implementations, however, sample the first bit on the first edge. The option R&S RTO6-K510 attempts to automatically detect this situation. Unfortunately, it is possible to trick the algorithm with waveforms that contain many (legitimate) violations. In these situations, the user can force a "First Edge" or "Second Edge" convention for handling edges. Edge sampling according to the "First Edge" convention is more likely to appear in biphase Manchester, but the software also supports this setting for ternary Manchester situations.

Bit Rate

Typically, a single bit rate is clearly specified in Manchester protocols; however, some implementations use a variable bit rate. By default, R&S RTO6-K510 automatically determines the bit rate with no user input required. However, there are fundamental ambiguities possible in Manchester, if the bitrate is unknown. In particular, sequences like "0000", "1111", "0101", "1010", and many situations involving "H" and "L" violations, cannot be decoded without a known bit rate. The situation becomes even less defined

with eventual Manchester coding violations. In these situations, a fixed "Bit Rate" setting has to be provided by the user to bypass the software's estimation algorithm.

13.6.2 Custom: Manchester / NRZ configuration

If you need information on how to get started with configuring the custom serial bus setup, see Chapter 13.6.2.6, "Configuring custom Manchester / NRZ signals", on page 566. Otherwise proceed with the configuration settings.

13.6.2.1 Custom: Manchester / NRZ configuration settings

Access: [Bus] > "Custom" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.

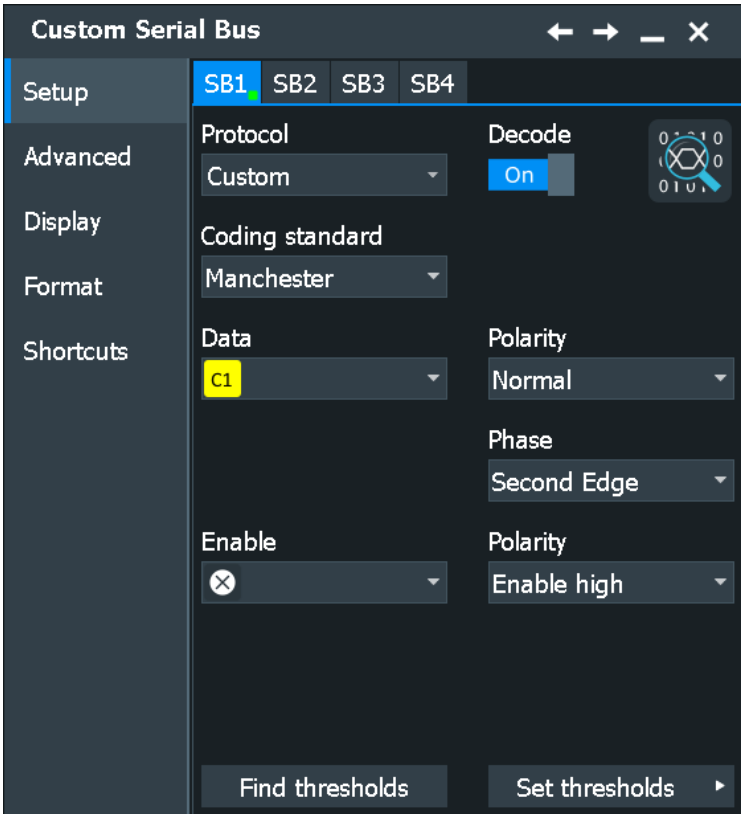


Figure 13-11: Coding standard selection in the serial bus protocol configuration dialog

See also: Chapter 13.1.1, "Setup - general settings", on page 485

Coding Standard
To define the coding of the custom serial bus to be analyzed, select one of the following standards:

"Manchester"	Selects the coding standard Manchester. Optional "Timing settings" are "Bit Rate" (default: disabled, 10 Mbit/s) and "Gap time" (default: disabled, 10 µs).
"Manchester II"	Selects the coding standard Manchester II, which is the inverted signal of the coding standard Manchester. Optional "Timing settings" are "Bit Rate" (default: disabled, 10 Mbit/s) and "Gap time" (default: disabled, 10 µs).
"NRZ Clocked"	Selects the coding standard NRZ Clocked.
"NRZ Unclocked"	Selects the coding standard NRZ unclocked.

Remote command:
`BUS<m>:CMSB:CODing` on page 1753

Data Source
Defines the input source for the custom serial bus data signal.

The data source for Manchester and NRZ coding standards is selected separately, independent of each other. The data source is set to default upon switching the coding standard.

Permitted source selections are:
- For "Manchester"/ "Manchester II":
  - Decoding: the analog, mathematical, and reference channels
  - Triggering: the analog channels
- For "NRZ Clocked":
  - Decoding: the analog, mathematical, reference and digital channels
    Digital channels can be only used if MSO option R&S RTO6-B1 is installed.
    Digital and analog channels cannot be used at the same time.
  - Triggering: the analog and digital channels
- For "NRZ Unclocked":
  - Decoding: the analog, mathematical, reference and digital channels
    Digital channels can be only used if MSO option R&S RTO6-B1 is installed.
    Digital and analog channels cannot be used at the same time.
  - Triggering: the analog channels

Remote command:
`BUS<m>:CMSB:MANChester:DATA` on page 1754
`BUS<m>:CMSB:NRZ:DATA` on page 1756

Clock Source
Defines the source for the custom serial bus clock signal (only available for the coding standard "NRZ Clocked").

Permitted source selections are the analog, mathematical, reference and digital channels.

Digital channels can be only used if MSO option R&S RTO6-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on the serial bus, when the NRZ clocked coding standard is selected, analog or digital channel sources are required.

Remote command:
`BUS<m>:CMSB:NRZ:CLCK` on page 1756

Data Polarity (Manchester)
Defines the polarity of the custom serial bus data signal in Manchester coding standards. The available settings are "Normal" or "Inverted".

Remote command:
`BUS<m>:CMSB:MANChester:POLarity` on page 1754

Data Polarity (NRZ)
Defines the polarity of the custom serial bus data signal in NRZ coding standards. The available settings are:

"Active high" The value "1" is represented by a voltage above the threshold.

"Active low" The value "1" is represented by a voltage below the threshold.

Remote command:
`BUS<m>:CMSB:NRZ:POLarity` on page 1759

Data Idle Polarity (NRZ Unclocked)
Defines the idle polarity of the custom serial bus data signal (only available for the coding standard "NRZ Unclocked"). The available settings are:

"Idle low" The base value of the clock is "0"; after an idle period, the data signal starts with a low-to-high transition.

"Idle high" The base value of the clock is "1"; after an idle period, the data signal starts with a high-to-low transition.

Remote command:
`BUS<m>:CMSB:NRZ:IDLPolarity` on page 1757

Clock Polarity (NRZ Clocked)
Defines the polarity of the custom serial bus clock signal (only available for the coding standard "NRZ Clocked"). The available settings are:

"Idle low" The base value of the clock is "0".

"Idle high" The base value of the clock is "1".

Remote command:
`BUS<m>:CMSB:NRZ:CPOLarity` on page 1757

Clock Phase (Manchester)
Defines the phase of the custom serial bus clock signal for the Manchester coding standards. The available settings are:

"Auto Detect" Lets the decoder automatically select the method ("First Edge" or "Second Edge") for detecting the clock phase.

"First Edge"
- At "Idle" = "low": data are captured on the clock's rising edge (low-to-high transition) and propagated on a falling edge.
- At "Idle" = "high": data are captured on the clock's falling edge (high-to-low transition) and propagated on a rising edge.

"Second Edge"
- At "Idle" = "low": data are captured on the clock's falling edge (high-to-low transition) and propagated on a rising edge.
- At "Idle" = "high": data are captured on the clock's rising edge (low-to-high transition) and propagated on a falling edge.

Note: The requirement to specify "First Edge" or "Second Edge" (or let the decoder decide) has the following background: In Manchester coding, an edge is always a transition from high to low (0) or from low to high (1). However, if the signal comes from the idle state, this implies that right before the first valid edge, there is always an overhead transition from idle to high or from idle to low. Some standards may regard this as a valid transition. To avoid a potentially ambiguous situation, a decision has to be made if the first edge is indeed only some overhead transition - or a transition that needs to be sampled. For more details on edge conditions, see Chapter 13.6.1.1, "Special features of Manchester coding", on page 547.

Remote command:
`BUS<m>:CMSB:MANChester:CPHase` on page 1756

Clock Phase (NRZ Clocked)
Defines the phase of the custom serial bus clock signal for the coding standard "NRZ Clocked", depending on "Clock Polarity". The available settings are:

"First Edge"
- At "Idle" = "low": data are captured on the clock's rising edge (low-to-high transition) and propagated on a falling edge
- At "Idle" = "high": data are captured on the clock's falling edge (high-to-low transition) and propagated on a rising edge

"Second Edge"
- At "Idle" = "low": data are captured on the clock's falling edge (high-to-low transition) and propagated on a rising edge
- At "Idle" = "high": data are captured on the clock's rising edge (low-to-high transition) and propagated on a falling edge

Remote command:
`BUS<m>:CMSB:NRZ:CPHase` on page 1758

Enable Source (NRZ)
Defines the input source for the custom serial bus enable signal.

If an input is chosen, signals are only decoded when this channel is in the enabled state. This allows you to mark a time when the signal on the selected source is active and when not.

Permitted source selections are the analog, mathematical, and reference channels.

When the serial bus trigger has been selected, the only permitted source selections are the analog channels "C1" – "C4", which are required for triggering.

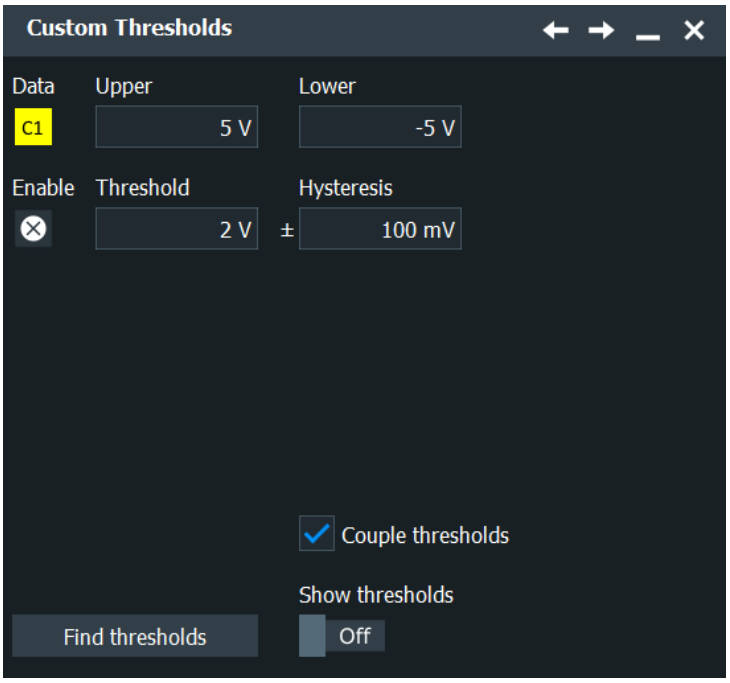
Math and Reference channels can only be selected, if no serial bus trigger is selected.

Remote command:
`BUS<m>:CMSB:NRZ:ENBLe` on page 1758

Enable Polarity (NRZ)
Selects whether the transmitted enable signal is active when the voltage is below the Set Thresholds ("Enable low") or higher than it ("Enable high").

Remote command:
`BUS<m>:CMSB:NRZ:ENAPolarity` on page 1759

Set Thresholds
Sets the threshold value for the digitization of each signal line and the hysteresis. If the signal voltage on the line is higher than the upper threshold, the signal state is high. Otherwise, if the signal voltage is below the lower threshold, the signal state is considered low.



- Manchester coding standards use 3-state signals with an upper and a lower voltage threshold in the range of -25 V to +25 V. A low-to-high transition requires the signal to exceed the upper threshold; a high-to-low transition requires the signal to fall below the lower threshold.
  You can set a hysteresis only for the enable signal.
- NRZ coding standards use a single voltage threshold for the data line. The value is in the range of -25 V to +25 V. It is entered into the middle of three available threshold input fields, or into the upper available threshold input field for NRZ Unclocked.
  You can set the hysteresis for all signal lines.
- In the NRZ Clocked coding standard, there is an additional clock voltage threshold available. This value in the range of -25 V to +25 V is entered into the upper threshold input field.
  You can set the hysteresis for the clock line.

There are four ways to set the threshold:

"Threshold" Directly sets the threshold values.

- For Manchester: upper threshold in the upper field, lower threshold in the lower field.
- For NRZ Clocked: clock threshold in the upper field, data threshold in the middle field and enable threshold in the lower field. Hysteresis values are in the right column.
- For NRZ Unclocked: data threshold in the upper field and enable threshold in the lower field. Hysteresis values are in the right column.

Remote command:
`BUS<m>:CMSB:MANChester:THReshold:HIGH` on page 1754
`BUS<m>:CMSB:MANChester:THReshold:LOW` on page 1754
`BUS<m>:CMSB:NRZ:THReshold:CLCK` on page 1759
`BUS<m>:CMSB:NRZ:THReshold:DATA` on page 1760
`BUS<m>:CMSB:NRZ:THReshold:ENBLe` on page 1760
`BUS<m>:CMSB:NRZ:HYSTeresis:CLCK` on page 1759
`BUS<m>:CMSB:NRZ:HYSTeresis:DATA` on page 1760
`BUS<m>:CMSB:NRZ:HYSTeresis:ENBLe` on page 1760

"Find thresholds" Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
`BUS<m>:SETReflevels` on page 1671

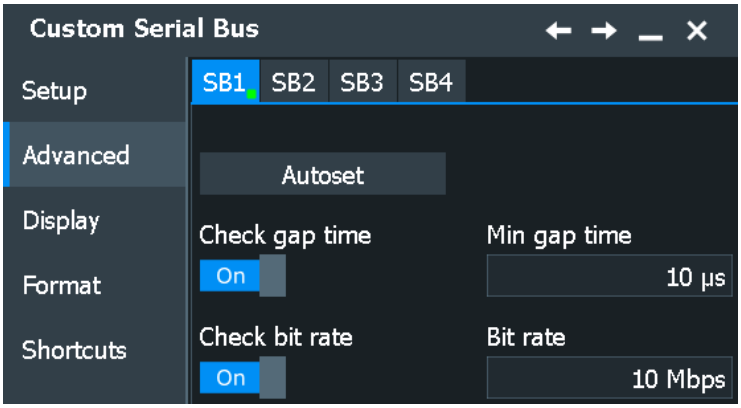
"Couple thresholds"

- For Manchester and Manchester II coding, the upper and lower thresholds are coupled to voltage values with the same magnitude but opposite sign (positive for the upper threshold and negative for the lower threshold). However, if the upper threshold is set to a negative voltage or the lower threshold is set to a positive voltage, coupling is disabled, and the other voltage (the one that was not actively set) is automatically adjusted, to avoid an upper threshold below the lower one, or a lower threshold above the upper one.
- For NRZ Clocked coding, the clock and data threshold values are coupled to the same voltage.

Remote command:
`BUS<m>:CMSB:MANChester:THReshold:COUPling` on page 1755
`BUS<m>:CMSB:NRZ:THReshold:COUPling` on page 1761

13.6.2.2 Advanced

Access: [Bus] > "Advanced" tab.



Autoset
Starts software algorithms for determining the signal threshold levels and bitrate.

Remote command:
`BUS<m>:FAUToset` on page 1671

Check bit rate
Enables the bit rate settings for the coding standards "Manchester" and "Manchester II". This setting is not available for "NRZ Clocked", but always enabled for the coding standard "NRZ Unclocked", and also for triggering on signals in any coding standard.

Remote command:
`BUS<m>:CMSB:BITRate:ENABle` on page 1762

Bit Rate
Defines the transmission speed setting for the data signal. A bit rate definition is optional for the coding standards "Manchester" and "Manchester II", not available for "NRZ Clocked", but obligatory for "NRZ Unclocked" (and also for triggering on signals in any coding standard). Default bit rate is 10 Mbps, permitted bit rates range from 300 bps to 2 Gbps.

For more details on the bit rate, see Chapter 13.6.1.1, "Special features of Manchester coding", on page 547.

Remote command:
`BUS<m>:CMSB:BITRate:VALue` on page 1762

Check gap time
Enables the gap time settings (always enabled for the coding standard "NRZ Unclocked", and also for triggering on signals in any coding standard).

Remote command:
`BUS<m>:CMSB:GAPTime:ENABle` on page 1762

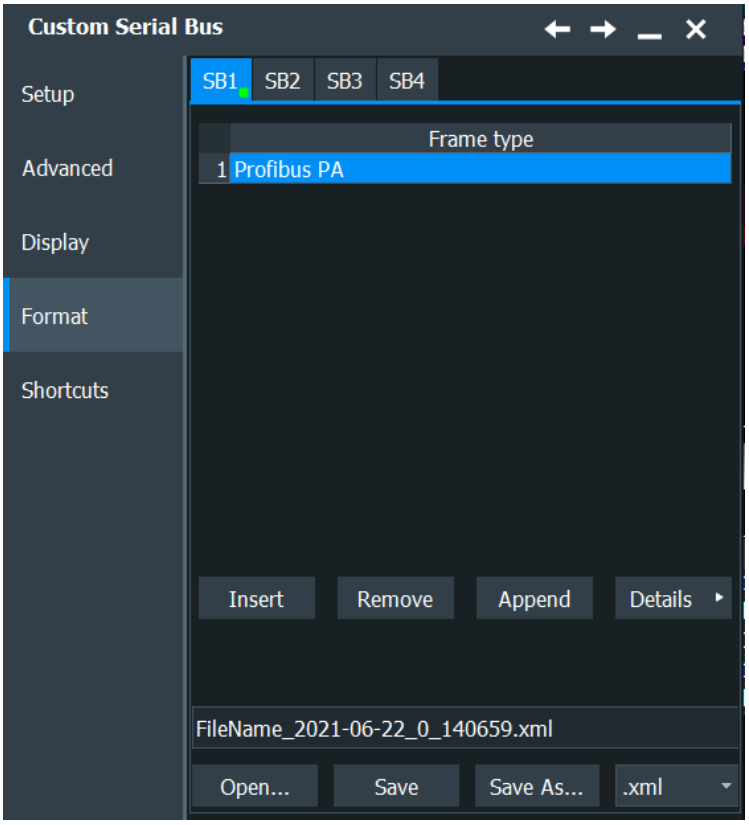
Min gap time
Specifies a minimum gap time (idle time or timeout) between two frames. A gap time definition is optional for the coding standards "Manchester", "Manchester II" and "NRZ Clocked", but obligatory for "NRZ Unclocked" (and also for triggering on signals in any coding standard). Default gap time is 10 µs, permitted gap times range from 1 ns to 1 s.

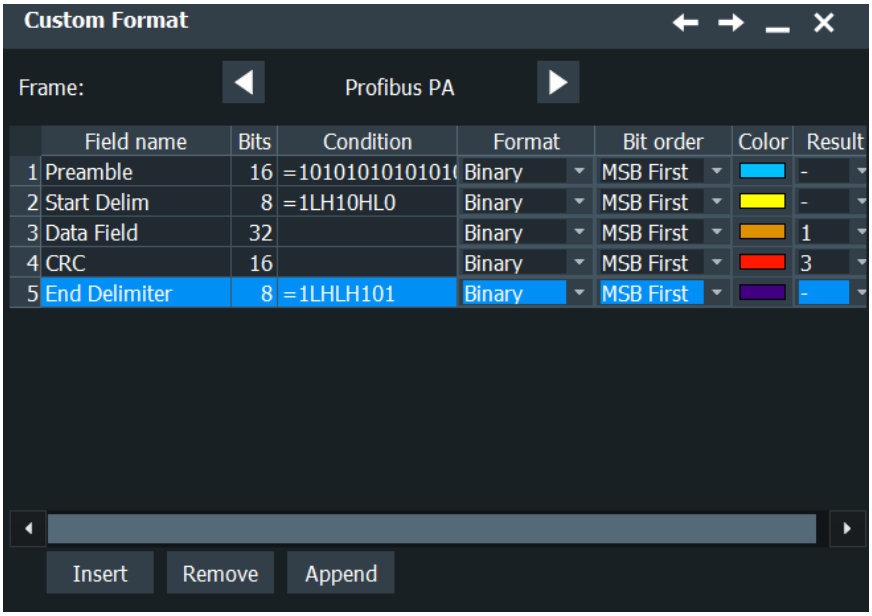
For more details on gap time and idle conditions, see Chapter 13.6.1.1, "Special features of Manchester coding", on page 547.

Remote command:
`BUS<m>:CMSB:GAPTime:VALue` on page 1762

13.6.2.3 Frame format configuration

This dialog enables you to describe the generic format and logical structure of typical protocols by creating customized frame descriptions of various structures and lengths.





Frames

A frame format description (or *frame description*, for short) is represented by one "page" in this dialog. It can be created by clicking on "Add". If one or several frame descriptions already exist, the new frame description is then created at the end of the frame format description list (or *frame list*, for short). Describing a frame format requires assigning it a name as well as creating Frame format configuration and specifying cell descriptions. The number of frame descriptions is limited to 100. The frame description that is on display can be deleted from the frame list by clicking "Remove".

The "Frame type" string is intended for the user to label the frame description (typically according to the specifications of the applicable protocol standard). For example, MDIO (Management Data Input/Output) specifies the frames "READ", "WRITE", "ADDRESS".

The frame format dialog also provides the features "Open", "Save", "Save As...", and "Explore...", to store created sets of frame descriptions into files (in ".xml" format), or load existing files of this kind.

The frame identification is executed top down, in the order in which the frame formats are described in the frame list. This provides a hierarchy of criteria for identifying frames.

In case it is required to change the order of previously created frame descriptions, it is recommended to save the set of frame descriptions, and then edit the XML file with any suitable editor. This also allows - with due care - additional editing features, if required.

If no user-defined frame description should be suitable to identify an incoming frame, per default such a "missed" frame is reproduced as "Undescribed Bits" in the honeycomb display. These bits are not shown in the results table.

Note: There may be frame descriptions that positively identifies each kind of frame, e.g. if no **equal** operator is defined for any of the fields. This "catches" every frame, even if there are other frame descriptions to follow in the frame list. Therefore, if a "catch all" frame description is used, it should be placed at the end of the frame list, or

it overwrites any subsequent frame description. However, instead of using a "catch all" frame description, the built-in "undescribed bits" display as mentioned above may be the better approach to create frame descriptions.

It is in the responsibility of the user to define unambiguous settings for each frame type. For a description of these conditions in `XML` file format and the required `XML` grammar and syntax, see Chapter 13.6.2.4, "XML syntax", on page 563.

Remote command:
`BUS<m>:CMSB:FRAMe<n>:TYPE` on page 1763
`BUS<m>:CMSB:ADDFrame` on page 1763
`BUS<m>:CMSB:CLR` on page 1763
`BUS<m>:CMSB:FCOunt?` on page 1763

Field name
A field description (which is represented by one row in one frame description) can be created at any position of a frame description by clicking on "Insert". This brings up a new field description in the active frame description, on top of the selected position. The "Append" button adds a cell description at the end of a frame description, below the lowest existing cell description. The number of Field descriptions is not limited. The "Remove" button deletes a selected field description from the active frame description.

The field result can be selected to be displayed in a specified result column of the decode table (see "Result" on page 563).

Note: The cell descriptions must be sequential and complete. No gaps are allowed, since the "Bis" are used to calculate the start position of the next cell.

A frame type is identified when all user-defined cell conditions are met, which can be regarded as related by the Boolean AND operator.

This can also locate a synchronization pattern, specified by the equal operator in the field. For example, if you define a "Preamble" cell with the condition *=FFFFFFFF*, the decoder scans the data for this pattern, and then synchronize to it.

The field in a frame are described by:

- Field Name
- "Bits" on page 559
- "Condition" on page 559
- "Format" on page 562
- "Color" on page 563
- "Result" on page 563

Remote command:
`BUS<m>:CMSB:FRAMe<n>:CCOunt?` on page 1764
`BUS<m>:CMSB:FRAMe<n>:APPend` on page 1764

Field Name
The strings in the column title describe cell names. They do not have to be unique; cell names are just for user support.

Remote command:
`BUS<m>:CMSB:FRAMe<n>:CELL<o>:NAME` on page 1764

Bits

This crucial information defines the length of the cell and - based on the previous cells - also the cell end position and the next cell start position within a frame.

If, for a given bit count, the Condition value is longer, it is truncated. If the condition value is shorter, it is padded with 0. Both truncation and padding occur at the left side of the condition value.

Examples:

- if "Condition" is "=111000" and the "Bit order" is "MSB", then
  - if "Bits" is 4, the truncated condition is "=1000"
  - if "Bits" is 8, the padded condition is "=00111000"
- if "Condition" is "=111000" and the bit order is "LSB" (accordingly, the condition in "MSB" format would be "=000111"), then
  - if "Bit s" is 4, the truncated condition is "=1000" for LSB and "=0001" for MSB
  - if "Bits" is 8, the padded condition is "=00111000" for LSB and "=00011100" for MSB

Remote command:

`BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITCount` on page 1764

Condition

This text field is used to apply various conditions and functionalities for a cell. Among others, it can be used to identify mandatory values (such as CRC checksum or ID) that help to identify a frame. The numeric format and bit order of the condition value has to match up with what is defined in the fields "Format" and "Bit order".

The following conditions are implemented:

"= (equal)" The **equal** operator (represented by the "=" sign) defines a pattern for the field to match. Valid condition entries are characters that match the cell's defined "Format", "Bit order", and "Bits". In binary format, for example, valid characters are "1", "0", "H" (high), and "L" (low).

Three cases have to be distinguished (cases A, B1, and B2), depending on the presence of a Variable Length Array: [] in the same frame description:

- **Case A**: If there is **no** "Variable Length Array" field, then each field marked with the equal operator acts as a key to identify a frame type. Only if all these fields match up with the expected value, the frame type is identified.
- **Cases B1 and B2**: If there **is** a "Variable Length Array" cell, then the equal operator has two different functionalities, depending on the position of the equal-operator cell within the frame description:
  - **B1**: If the cell is located *anywhere before* the "Variable Length Array" cell, the condition acts as a key to identify a frame type (as in case A).
  - **B2**: If the cell is located *immediately after* the "Variable Length Array" cell, the condition acts as an array delimiter.
    (Note: If the cell, which is marked with the equal operator, is located after the "Variable Length Array" cell, but *not* immediately after it, the decode result is unpredictable.)

Typically, Manchester protocols use code violations for synchronization. The states "H" and "L", supported by the equal operator in binary Format, mark that a transition is expected at this bit, but only a high or low signal is found.
Examples for the MVB protocol:
Master - Delimiter: "=*1LH0LH000*"
Slave - Delimiter: "=*0000LH0LH*"
For more details on the violation symbols "H" and "L", see Chapter 13.6.1.1, "Special features of Manchester coding", on page 547.
Also, the length of the pattern must correspond to the bit field length (or the results are unpredictable).

"[ ] (array)" The **array** operator (represented by the "[" and "]" bracket signs) defines the number of permissible repetitions of the cell.

Example: Fixed-Length Array: [n]

The length parameter "n" is a decimal number > 0, which determines that the cell is repeated n times within the frame. If, for example, the Bits is 8, then the array operator identifies n cells of 8-bit length, and present them in the results table and honeycomb display with the specified name and color.

A fixed-length array is treated the same as other cells, except the real length of such an array is n · bit count.

Example: Variable Length Array: []

This array operator with empty "[" and "]" bracket signs does not determine a fixed size array. The cell could be repeated any number of times, including 0 times. As a result, the cell and the frame are of unspecified length (a situation that covers typical use cases).

The length of the frame is then determined by the end of frame condition, which can be an operator or a gap.

For processing reasons, only one (1) variable length array is supported in a frame, and a delimiter must follow immediately in the next cell after it. This is required to enable the software to correctly terminate the array. Otherwise the bits could not be assigned correctly, and it would not be possible to determine where a repletion starts and where it ends. With an end of frame condition, the software can calculate the length of one single array within a frame. But if there were more arrays, it would be impossible to know which array was how long.

The variable length array can also be the last cell of a frame. In this case, no delimiter is required. If decoded successfully, the detailed view in the results table shows the elements of the array. If the cell name of the array is "Data", then the detail view of result table displays the elements with an array index as "Data: 1", "Data: 2" ... etc.

If the variable length array cell is selected, it is shown as array [n], where n is the actual size detected in the waveform.

As an exception to the rule, it is permissible to assign the variable length array to consecutive fields. In this case, the fields are treated as a structure which is repeated. For example, if two consecutive fields are defined as A[] and B[], the decoder creates a sequence of ABABAB until the end condition has been detected.

Note: It is possible to combine check functions in a dynamic array. In the example above, if B[] is extended by odd(1), with "1" being the index of A[], then B checks the parity for each index of A.

"crc5usb(n-m)" The **crc 5-bit** operator performs a check for a 5\-bit CRC function using the polynomial as defined by the USB standard. n and m define the index range for the CRC check.
For example, if the CRC shall check fields 1 to 4, the function shall be written "crc5usb(1-5)".
If the range of the CRC check includes an array, all elements in the array is included in the CRC check.
If the check fails, the CRC field is marked as "CRC error" in the result details and displayed in the color red in the honeycomb display. The frame that contains the field is marked in the same way, except if another higher priority error is found within this frame.

"odd(n-m), even(n-m)" The "**parity**" operators perform checks on odd or even parity in the given index range n to m.
Odd parity is fulfilled if the count of "1" bits in the range including the parity bit is odd. Even parity is fulfilled if the count of "1" bits in the range including the parity bit is even.
If the parity check fails, the parity field is marked as "CRC error" in the result details and displayed in red color in the honeycomb display.
The frame that contains the field is marked in the same way, except if another higher priority error is found within this frame.

"{Block}" The **block** operator is represented by the "{" and "}" bracket signs. Consecutive fields marked with "{Block}" and using the same name are displayed in the honeycomb display as a consecutive packet of name "Block" with the first field's color. This feature is a visual effect in the honeycomb only.

Remote command:
`BUS<m>:CMSB:FRAMe<n>:CELL<o>:CONDition` on page 1765

Format
Selects from the following numeric data formats for the Condition value:
- Decimal
- Hexadecimal
- Octal
- Binary

The following rules apply:
- If the condition value contains at least one "H", "h", "L", "l", "X", "x", and the remaining characters only contain "1" and/or "0", the numeric format is automatically interpreted as binary, regardless of its definition.
- The wild-card characters "x" and "X" are only supported in binary format.
  Examples: If the numeric format is set to be "HEX", then:
  – "=1HL111000" is valid (read as binary)
  – "=0x10101" is valid (read as binary, the "x" is interpreted here as a wild card)
  – "=1010" is valid (read as HEX, with a total of 16 bits)
  – "=0x5A" is valid (read as HEX "5A", since "0x" is a valid HEX prefix; nevertheless, it is recommended to enter "5A" instead)
  – "=5X12" is invalid
  – "=1H33" is invalid

Remote command:
`BUS<m>:CMSB:FRAMe<n>:CELL<o>:FORMat` on page 1765

Bit Order
This defines, in which order the bits of a cell's Condition value are evaluated: either the most significant bit (MSB) or the least significant bit (LSB) first. Since the bit order is considered for the interpretation of the condition, you should specify MSB or LSB correctly.

Remote command:
`BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITorder` on page 1766

Color
Opens a dialog to select the color representation of different cells in the honeycomb display. Assigning user-selected colors helps to interpret the decode results more easily.

In the "Adjust Colors" dialog, you can either select one of the predefined colors or define a new one.

Remote command:
`BUS<m>:CMSB:FRAMe<n>:CELL<o>:CRGB` on page 1766

Result
This determines which cells shall be displayed in which result columns of the decode table. No index means that the result is not displayed. The decode table supports three result columns, which have to be unique for each frame type. For different frame types, though, the user can define different result columns, to display unrelated information.

Note: To see more than the three selected results, bring up a full list of the states and values of all cells by activating "Show details" in the decode table dialog.

Remote command:
`BUS<m>:CMSB:FRAMe<n>:CELL<o>:CLMN` on page 1766

Open or Save XML File
For efficient working and for convenient exchange of frame descriptions, they can both be loaded ("Open") or saved ("Save" / "Save As...") in `XML` file format. "Explore..." opens the `SaveXML` folder, which is the "Default Path" for saving frame descriptions.

Remote command:
`BUS<m>:CMSB:LOAD` on page 1767
`BUS<m>:CMSB:SAVE` on page 1767

13.6.2.4 XML syntax

This chapter explains the required grammar and syntax of `XML` files, which contain custom frames and can be loaded or saved. Below is a typical example of such an `XML` file:

```
<?xml version="1.0" encoding="utf-8"?>
<FrameDescription xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" Version="1" xsi:noNamespaceSchemaLocation=".\Schema\FrameDescription.xsd">
  <Frame Type="00">
    <Format Name="Start Delim" BitCount="16" Condition="=101010101HL" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="ffff00" Column="-"/>
    <Format Name="OP-00" BitCount="8" Condition="=00000000" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="0080ff" Column="1"/>
    <Format Name="data" BitCount="8" Condition="[]" NumericFormat="Hex" BitOrder="MSB First" ColorRGB="10a500" Column="2"/>
    <Format Name="End Delim" BitCount="8" Condition="=1HLHL101" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="ffff00" Column="-"/>
  </Frame>
  <Frame Type="01">
    <Format Name="Start Delim" BitCount="16" Condition="=101010101HL" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="ffff00" Column="-"/>
    <Format Name="OP-01" BitCount="8" Condition="=00000001" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="0080ff" Column="1"/>
    <Format Name="data" BitCount="8" Condition="[]" NumericFormat="Hex" BitOrder="MSB First" ColorRGB="10a500" Column="2"/>
    <Format Name="CRC" BitCount="16" Condition="" NumericFormat="Hex" BitOrder="MSB First" ColorRGB="ff808080" Column="2"/>
    <Format Name="End Delim" BitCount="8" Condition="=1HLHL101" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="ffff00"" Column="-"/>
  </Frame>   <Frame Type="ff">
    <Format Name="Start Delim" BitCount="16" Condition="=101010101HL" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="ffff00" Column="-"/>
    <Format Name="OP-ff" BitCount="8" Condition="=11111111" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="0080ff" Column="1"/>
    <Format Name="data" BitCount="8" Condition="[]" NumericFormat="Hex" BitOrder="MSB First" ColorRGB="ffb233af" Column="2"/>
    <Format Name="CRC" BitCount="16" Condition="" NumericFormat="Hex" BitOrder="MSB First" ColorRGB="ff808080" Column="2"/>
    <Format Name="End Delim" BitCount="8" Condition="=1HLHL101" NumericFormat="Binary" BitOrder="MSB First" ColorRGB="ffff00" Column="-"/>
  </Frame>
</FrameDescription>
```

Figure 13-12: Example of XML file syntax with three custom frame format descriptions

The first out of three `XML` frames in Figure 13-12 is interpreted by the software in the following way:

A suitable `XML` file as shown in Figure 13-12 is composed as follows:

Header:

```
<?xml version="1.0" encoding="utf-8"?>
```

Root Element:

```
<FrameDescription xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 Version="1" xsi:noNamespaceSchemaLocation=".\Schema\FrameDescription.xsd">
```

The root element contains the "Frame Description" attributes, including a link for the file`FrameDescription.xsd`. This schema file, which is installed in the system, enables the software to validate an `XML` file before opening it.

Frame:

A frame description must include between *0* and *n* tags of the following kind:

```
<Frame> </Frame>
```

Frame Type:

Each "<Frame>" tag requires a "Type" attribute in string format:

```
<Frame Type = "string">
```

Tells the software the name of each frame.

Format:

Each frame must include between *1* and *n* tags of the following kind:

```
<Format> </Format>
```

Together with the attributes, this is written in short form, as in Figure 13-12:

```
<Format attribute... attribute... attribute... />
```

The format describes the fields in each frame. It can have the following attributes:

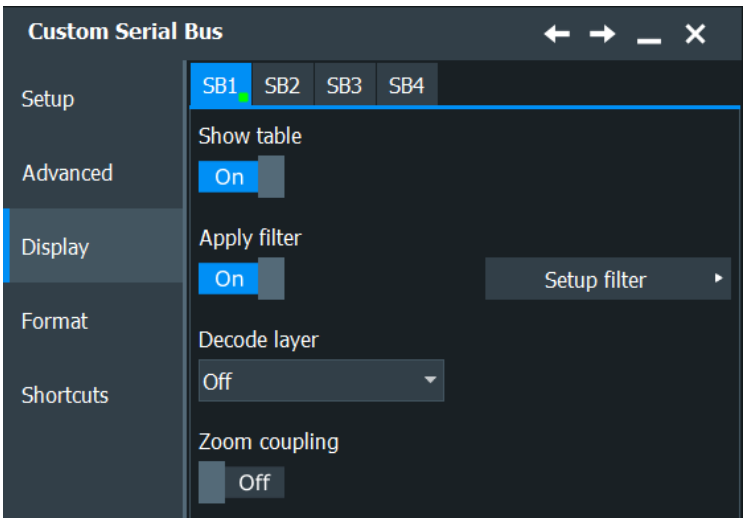
- "Field Name", in string format.
- "Bit count" specifies the length of bits, in numerical format.
- "Condition" (optional) identifies the bit pattern to match, in string format.
- "Format " allows the following choices:
  - "Decimal"
  - "Hex"
  - "Octal"
  - "Binary"
- "Bit order" allows two alternatives:
  - "MSB First" (most significant bit first)
  - "LSB First" (least significant bit first)
- "Color" allows you to set a user defined ARGB hexadecimal color value.
- The "Result Column" has four options:
  - "-" (none, which is the default)
  - "1"
  - "2"
  - "3"

13.6.2.5 Custom: Manchester / NRZ display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

To enhance the decode possibilities of the custom serial protocol, you can use an additional setting in the "Display" tab: "Decode layer".



Apply Filter
Enables filtering on a custom serial bus. For details see: Chapter 13.6.4, "Custom filter", on page 572.

Remote command:
`BUS<m>:CMSB:FILTer:ENABle` on page 1774

Decode layer
Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

"Final"	...
"Edges"	...
"Binary"	...
"Synchronization"	...

13.6.2.6 Configuring custom Manchester / NRZ signals

For configuration, assign the lines to the input channels, define the active states and the logical thresholds, and specify frame format descriptions.

Serial bus setup

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. In the "Setup" tab, select the protocol: "Custom".
4. Switch to the "Trigger" tab, tap "Source" and select "Serial bus".

   This prevents using digital waveforms (Math and Reference) as channel signals.

   Note: For triggering on a custom serial bus, analog input channels are required.
5. Switch back to the "Decode" tab.
6. Tap "Coding Standard" and select the coding ("Manchester", "Manchester II", "NRZ Clocked", or "NRZ Unclocked") you want to set up.
7. Select the polarity and phase of the data signal (and potentially of the clock signal).
8. Set the logical thresholds. See "Set Thresholds" on page 553.
9. Enable "Decode".
10. Switch to the "Format" tab and open or create frame format descriptions.

For details on configuration settings, see Chapter 13.6.2.1, "Custom: Manchester / NRZ configuration settings", on page 549.

13.6.3 Custom: Manchester / NRZ trigger

If you need information on how to get started with triggering on Custom serial bus signals, see Chapter 13.6.3.2, "Triggering on custom Manchester / NRZ serial bus", on page 572. Otherwise proceed with the Custom serial bus trigger settings.

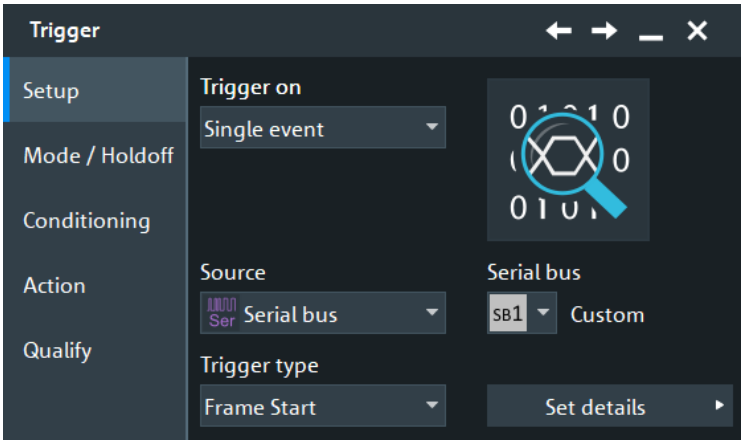
13.6.3.1 Custom: Manchester / NRZ trigger settings

Access: [Bus] > "Trigger" tab > "Source = Serial Bus" > select "Protocol = Custom"

In this section, all trigger settings are described. Their availability on the instrument depends on the selected coding standard and trigger type. The user interface of the instrument displays only appropriate settings and guides you through the trigger setup.

You can adjust the "Gap time" and "Bit rate" also in the "Trigger" tab if necessary.

For a list of supported trigger conditions, refer to the data sheet.



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Type

Defines the trigger type for custom serial bus analysis. The available trigger types are "Frame Start", "Pattern" and "Advanced".

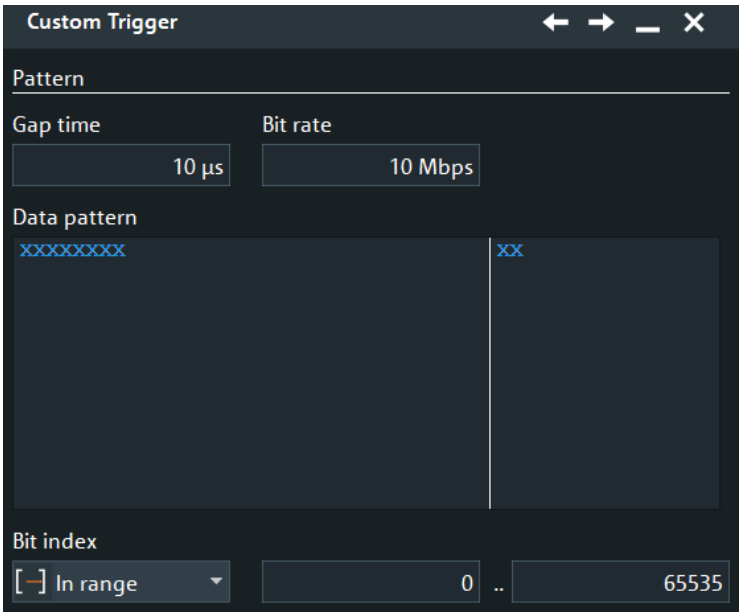
Remote command:

`TRIGger<m>:CMSB:TYPE` on page 1768

Frame Start ← Type
For Manchester and NRZ Clocked coding standards, the frame start trigger is set to the end of the gap time. The start of frame (SOF) condition is the first bit after the gap (timeout).

For the NRZ Unclocked coding standard, the trigger requires that the signal contains a start bit. The frame start trigger follows the gap time and is set to the end of the start bit.

Pattern ← Type
Specifies the pattern match conditions for a payload data check. The trigger is set to the first occurrence of a matching data bit pattern (which can be freely specified), starting after the minimum gap time, and after the detected start of the data frame. The trigger instant is after the last bit of the specified data pattern.



"Data Pattern"	Specifies the data pattern that is to be found and triggered. The pattern can be entered in binary or hexadecimal format, maximum pattern length is 256 binary characters or 64 hexadecimal characters.
"Bit index operator"	Sets the operator ("Equal", "Greater or equal", or "In range") to set a specific bit index (data position).
"Bit index"	Sets the bit index (data position), or the start value of a bit index range. Default bit index value is 0, permitted values range from 0 to 65535.
"Bit index to"	Sets the end value of a bit index range (data position range). Available only, if the "Bit index operator" is set to "In range". Default bit index end value is 65535, permitted values range from 0 to 65535.

"Word Length" Sets the number of bits in an NRZ Unclocked word (hence, the size of the data frame). Default word length is 8 bits, permitted lengths range from 0 bits to 31 bits .

Note: The NRZ Unclocked coding standard requires a signal that contains both a start bit and a stop bit:

- The start bit should be opposite in polarity to the idle state of the signal, and it is the first transition detected following the gap time.
- The stop bit should be the same polarity of the idle state, and it is the last bit in a data frame.

The end of the stop bit and the detection of the next frame's start bit constitutes the maximum gap time.

In order for the trigger to operate correctly, **you have to specify** the correct word length in the trigger menu. The trigger then counts the number of bits it decodes, and when the count matches the word length, the next bit is treated as the stop bit.

"Gap Time" Sets the minimum gap time for synchronization. The trigger is set to a position after the gap time, when the other trigger conditions are met. Default gap time is 10 µs, permitted gap times range from 1 ns to 1 s.

"Bit Rate" Sets the transmission speed for the data signal. Default bit rate is 10 Mbps, permitted bit rates range from 300 bps to 50 Mbps.

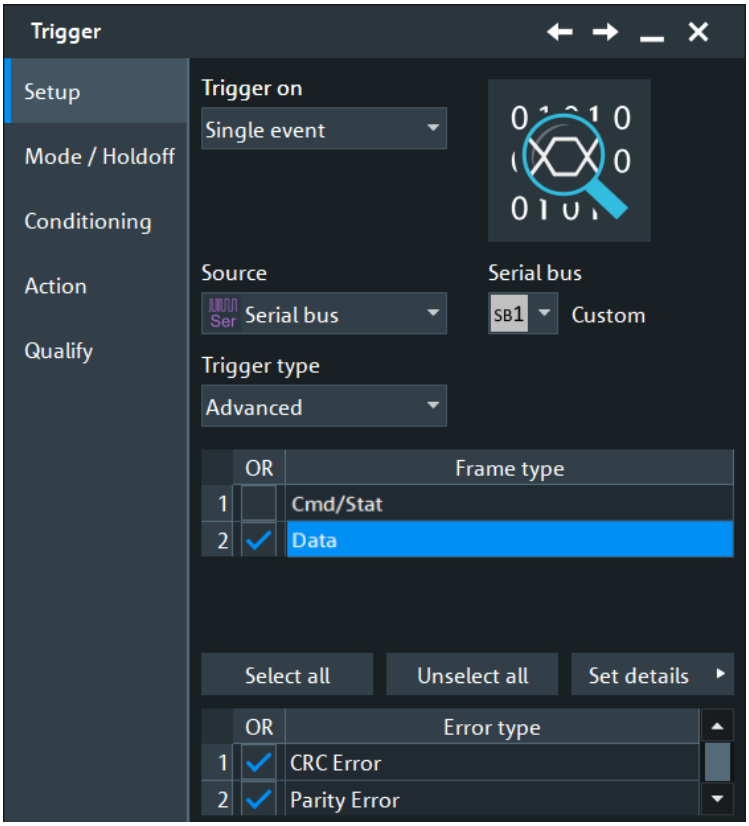
Remote command:
`TRIGger<m>:CMSB:PATTern` on page 1769
`TRIGger<m>:CMSB:ICONdition` on page 1769
`TRIGger<m>:CMSB:IMIN` on page 1769
`TRIGger<m>:CMSB:IMAX` on page 1770
`BUS<m>:CMSB:GAPTime:VALue` on page 1762
`TRIGger<m>:CMSB:NRZ:WRDLength` on page 1770

Advanced

Trigger on various frame types, fields in the frames, and data patterns.

Which settings are available, depends on what you have defined in the "Format" tab.

You can further refine the trigger criteria, selecting which frames you want to select for the triggering and what conditions their field values must fulfill.



Error types: Item name, Enable ← Advanced
The table lists the error types you can trigger on. Select the required errors in the "Enable" column. Available are "CRC" error and "Parity" error.

Remote command:
`TRIGger<m>:CMSB:ADVanced:ERENable` on page 1772
`TRIGger<m>:CMSB:ADVanced:ERRor<n>:ENABle` on page 1772

Frame type ← Advanced
Selects the frame type for the custom bus trigger analysis.

You can define individual checking parameters for the fields listed in the "Trigger type dependent settings".

To specify these parameters, select a field from this list and define the data and/or index operators and values, or the bit state.

The trigger instant is the last criterion that is fulfilled.

Remote command:
`TRIGger<m>:CMSB:ADVanced:FRAMe<n>:ENABle` on page 1770

Field name / Condition / Enable ← Advanced
This table lists the field numbers and names in the selected frame together with a summary of the user settings of checking conditions for each field and a checkbox to enable the checking.

Select a field in the table to specify the checking conditions for this field in the "Data", "BitState" and/or "Index" dialog (whichever applies). The condition is only applied, and the "Condition" is only shown in the table, if "Enable" is checked.

Remote command:
`TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:ENABle` on page 1770
`TRIGger<m>:CMSB:ADVanced:FIENable` on page 1770

BitState ← Advanced
Defines the bit state to be checked for the selected field. Permitted bit states are "1", "0" or "X" (do not care).

Remote command:
`TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:BIT` on page 1771
`TRIGger<m>:CMSB:ADVanced:BIT` on page 1771

Data ← Advanced
Defines for the selected field, how a data check is executed.

"Condition" Defining specific data or a data range requires to set the operator to one of the following conditions: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies data or sets the start value of a data range.

"Max" Sets the end value of a data range, if "Condition" is set to "In range" or "Out of range".

Remote command:
`TRIGger<m>:CMSB:ADVanced:DMAX` on page 1771
`TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMAX` on page 1771
`TRIGger<m>:CMSB:ADVanced:DMIN` on page 1772
`TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMIN` on page 1772
`TRIGger<m>:CMSB:ADVanced:DOPerator` on page 1772
`TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DOPerator` on page 1772

Index ← Advanced
Defines for the selected field, how an index check is executed.

"Condition" Defining a specific index or an index range requires to set the operator to one of the following conditions: equal, in range.

"Min" Specifies the index or sets the start value of an index range.

"Max" Sets the end value of an index range, if "Condition" is set to "In range".

Remote command:
`TRIGger<m>:CMSB:ADVanced:IMAX` on page 1773
`TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMAX` on page 1773
`TRIGger<m>:CMSB:ADVanced:IMIN` on page 1773
`TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMIN` on page 1773
`TRIGger<m>:CMSB:ADVanced:IOPerator` on page 1773
`TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IOPerator` on page 1773

13.6.3.2 Triggering on custom Manchester / NRZ serial bus

Prerequisite: A bus is configured for the custom serial bus signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [Bus] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to Custom.
5. Tap "Type" and select the trigger type to be used for custom serial bus protocol analysis.
6. Depending on the selected custom serial bus coding standard, more setup conditions have to be specified.

For information on how to proceed with the configuration settings, see Chapter 13.6.3.1, "Custom: Manchester / NRZ trigger settings", on page 567.

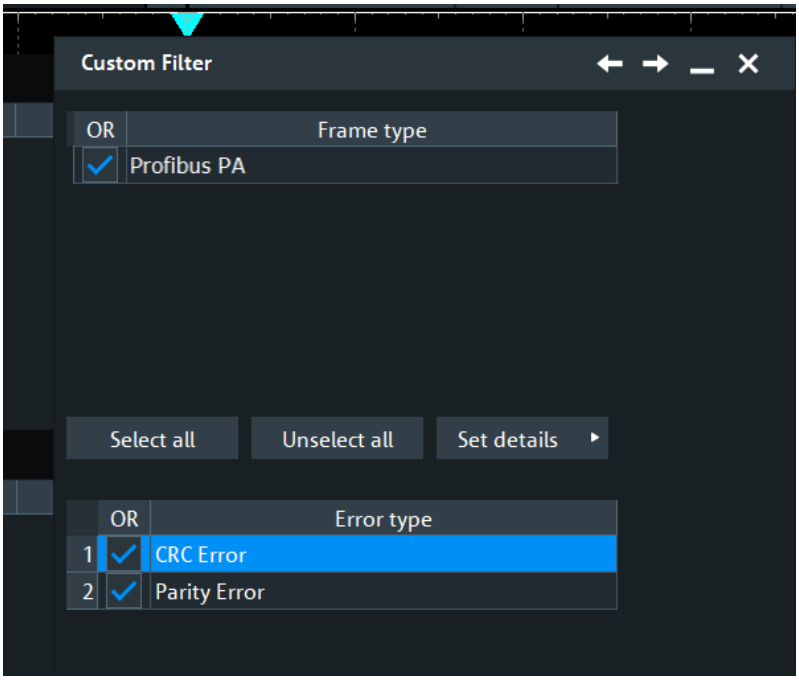
13.6.4 Custom filter

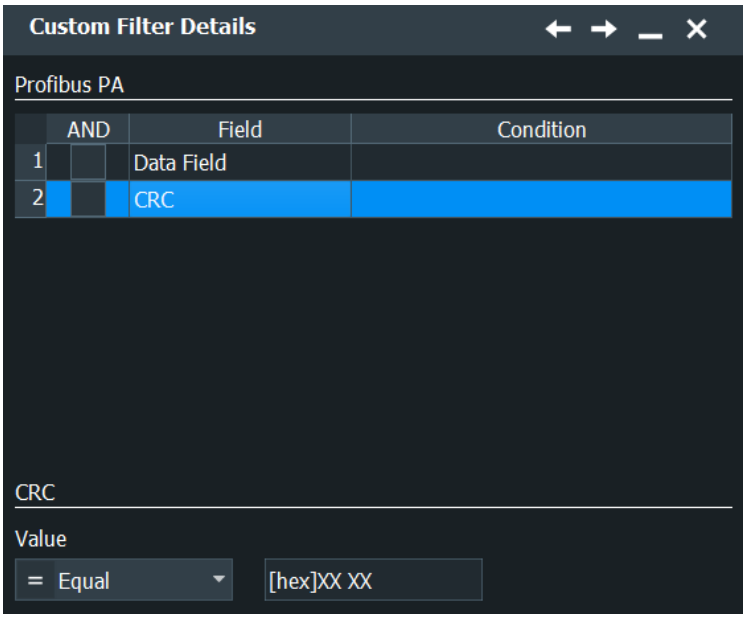
With the filter function you can display only selected events in the acquired data. Only one single filter can be enabled on the protocol at a time.

13.6.4.1 Custom filter settings

Access:

1. Press [Bus].
2. Set "Protocol = Custom".
3. In the "Display" tab, enable "Apply filter".
4. Open "Setup filter".





Frame type

Selects the frame type for the custom bus filtering.

You can define individual checking parameters for the fields listed in the "Trigger type dependent settings".

To specify these parameters, select a field from this list and define the data and/or index operators and values, or the bit state.

Remote command:

`BUS<m>:CMSB:FILTer:FRAMe<n>:ENABle` on page 1775

Error types: Item name, Enable
The table lists the error types you can filter on. Select the required errors in the "Enable" column. Available are "CRC" error and "Parity" error.

Remote command:
`BUS<m>:CMSB:FILTer:ERRor<n>:ENABle` on page 1774

Field name / Condition / Enable
This table lists the field numbers and names in the selected frame together with a summary of the user settings of checking conditions for each field and a checkbox to enable the checking.

Select a field in the table to specify the checking conditions for this field in the "Data", "BitState" and/or "Index" dialog (whichever applies). The condition is only applied, and the "Condition" is only shown in the table, if "Enable" is checked.

Remote command:
`BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:ENABle` on page 1777

Bit State
Defines the bit state to be checked for the selected field. Permitted bit states are "1", "0" or "X" (do not care).

Remote command:
`BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:BIT` on page 1775

Data
Defines for the selected field, how a data check is executed.

"Condition"	Defining specific data or a data range requires to set the operator to one of the following conditions: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.
"Min"	Specifies data or sets the start value of a data range.
"Max"	Sets the end value of a data range, if "Condition" is set to "In range" or "Out of range".

Remote command:
`BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:DMAX` on page 1776
`BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:DMIN` on page 1776
`BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:DOPerator` on page 1776

Index
Defines for the selected field, how an index check is executed.

This function is avalible only for array fields.

"Condition"	Defining a specific index or an index range requires to set the operator to one of the following conditions: equal, in range.
"Min"	Specifies the index or sets the start value of an index range.
"Max"	Sets the end value of an index range, if "Condition" is set to "In range".

Remote command:
`BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:IMAX` on page 1777
`BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:IMIN` on page 1777
`BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:IOPerator` on page 1777

13.6.5 Custom: Manchester / NRZ decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Serial Bus" dialog > "Setup" tab, enable "Decode".
2. In the "Serial Bus" dialog > "Display" tab, select additional result display settings: "Show table" and "Show binary".
3. If required, enable "Zoom coupling"

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

In contrast to other protocols, the instrument captures and decodes customizable serial bus signals according to the individual frame configuration.

Decode Results Table

The "Decode results" table contains information about all decoded frames.

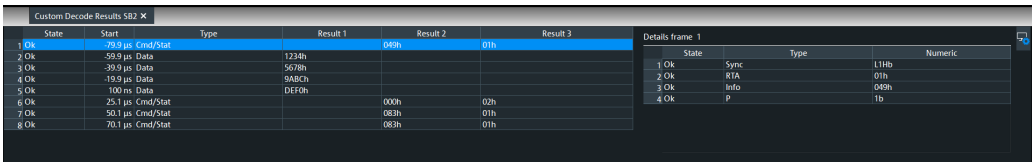


Table 13-8: Content of the "Decode results" table

Column	Description
State	Overall state of the frame: either OK or the relevant error condition (e.g. preamble, length)
Start	Start time of the frame
Type	Frame type as specified in the "Frame type" field of the "Frame Format" description dialog
Result 1	1st field content as specified in the "Result" of the "Frame Format" description dialog
Result 2	2nd field content
Result 3	3rd field content

Table 13-9: Content of the "Decode results details" table

Column	Description
State	Overall state of the cell: either OK or the relevant error condition (e.g. length error)
Type	The type of frame, as specified with the "Frame type"
Numeric	Data content of the cell (e.g. 0x1, 1000LL00L)

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands are described in Chapter 24.17.7.4, "Decode results", on page 1778.

13.6.6 Search on decoded custom Manchester / NRZ data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 11, "Search functions", on page 419.

If you need information on how to get started with searching custom serial bus data, see Chapter 13.6.6.3, "Searching custom serial bus data", on page 577. Otherwise proceed with the custom serial bus search setup.

13.6.6.1 Custom Manchester / NRZ search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"

Search criteria
Use the "Search criteria" dialog to define the event types to be searched. Available event types are "Frame" and "Error".

Individual search parameters, which do not depend on the coding standard and trigger settings of the custom serial bus, can be specified in the tabs below the "Search criteria" dialog.

Frame
Searches for four different frame conditions.

Remote command:
SEARch:TRIGger:CMSB:ERRor<m>:ENABle on page 1785
SEARch:TRIGger:CMSB:FIENable on page 1785
SEARch:TRIGger:CMSB:FRAMe<m>:ENABle on page 1787
SEARch:TRIGger:CMSB:FRENable on page 1785
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:BIT on page 1785
SEARch:TRIGger:CMSB:BIT on page 1785
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMAX on page 1786
SEARch:TRIGger:CMSB:DMAX on page 1786
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMIN on page 1786
SEARch:TRIGger:CMSB:DMIN on page 1786
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DOPerator on page 1787
SEARch:TRIGger:CMSB:DOPerator on page 1787
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMAX on page 1788
SEARch:TRIGger:CMSB:IMAX on page 1788
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMIN on page 1788
SEARch:TRIGger:CMSB:IMIN on page 1788
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IOPerator on page 1788
SEARch:TRIGger:CMSB:IOPerator on page 1788

Error Condition
Searches for two error conditions: "CRC Error" or "Parity Error".

Remote command:
SEARch:TRIGger:CMSB:ERRor<m>:ENABle on page 1785

13.6.6.2 Custom serial bus search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands are listed in Chapter 24.17.7.6, "Search results", on page 1789

13.6.6.3 Searching custom serial bus data

Prerequisite: A serial bus is configured for the custom serial bus signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Tap "Setup search".

3. If the dialog box does not contain a search entry, tap the "Add" icon to create one, as described in "To create a user-defined search" on page 434.
4. Specify search criteria according to Chapter 13.6.6.1, "Custom Manchester / NRZ search setup", on page 576.
5. To acquire a waveform, press [Single].

   The R&S RTO6 performs a custom serial bus decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).
6. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

   The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.7 CAN and CAN FD (option R&S RTO6-K520)

CAN is the Controller Area Network, a bus system designed by Bosch for use within automotive network architecture, for example, for brake, power train and engine management. Today, it is also used in many other systems, for example, in industrial machines, aerospace, subsea, merchant marine etc..

More than 20 years after the invention of CAN, communication needs have increased, and CAN has reached it's bandwidth limits in some application fields. Therefore, Bosch specified an improved CAN protocol with flexible data rate - CAN FD. It introduces a higher bit rate in the data phase up to 15 Mbit/s and an extended data field from up to 64 bytes.

The R&S RTO6 provides decoding, triggering and searching CAN and CAN FD signals with option R&S RTO6-K520.

13.7.1 The CAN protocol

CAN frame format and in CAN FD frame format can coexist within the same network. CAN FD shares the physical layer, with the CAN protocol as defined in the BOSCH CAN Specification 2.0. The frame format is different. There are two new control bits in the CAN FD frame, the first enabling the new frame format with different data length coding and the second optionally switching to a faster bit rate after the arbitration is decided. New CRC polynomials are introduced to secure the longer CAN FD frames.

13.7.2 CAN and CAN FD configuration

13.7.2.1 CAN and CAN FD configuration settings

Access: [Bus] > "CAN" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.

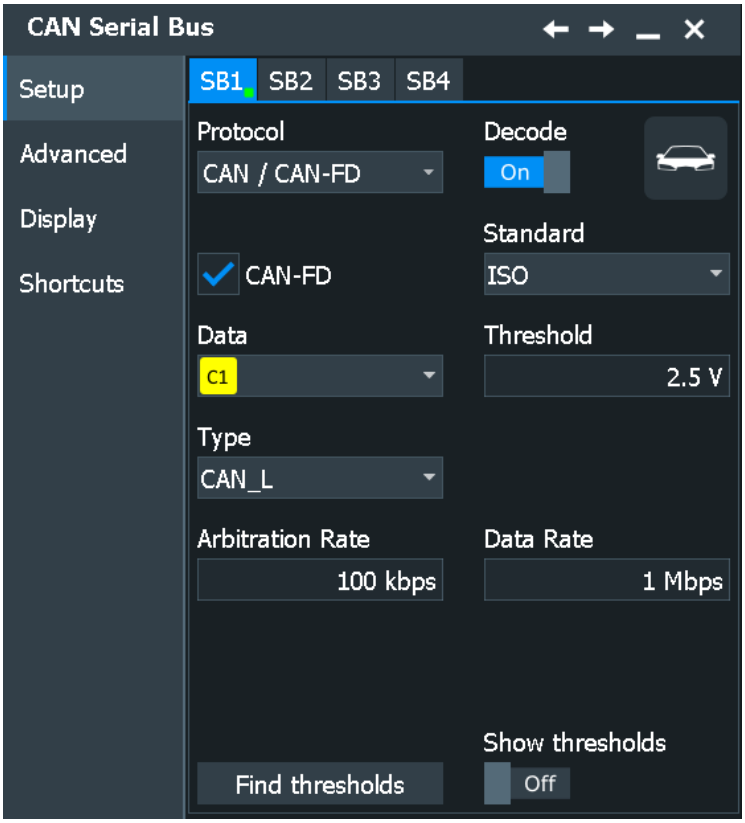


Figure 13-13: Configuration for CAN FD

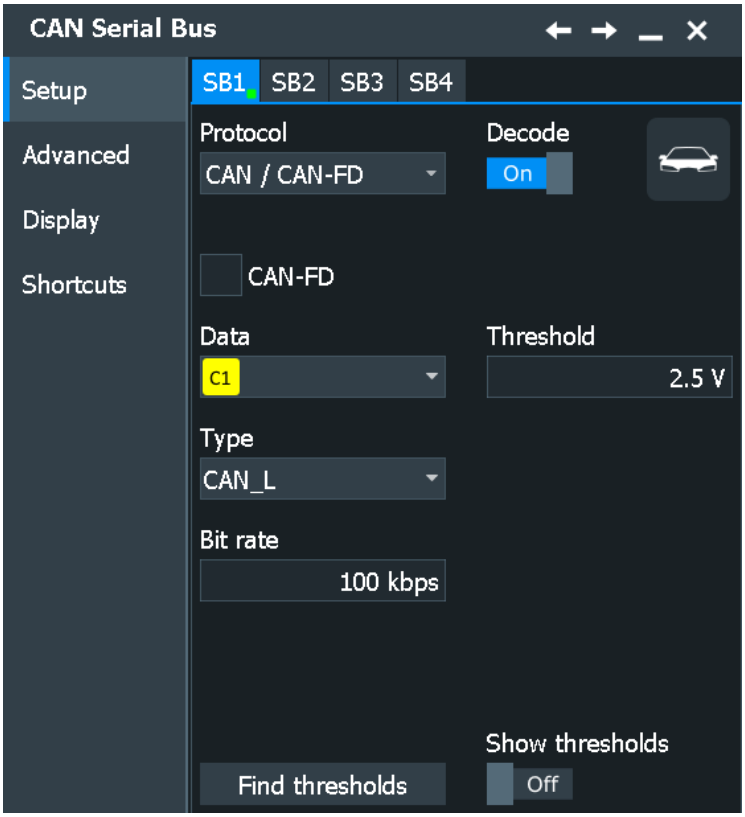


Figure 13-14: Configuration for CAN

See also: Chapter 13.1.1, "Setup - general settings", on page 485

Protocol
Defines the protocol type of the selected serial bus.

Remote command:
`BUS<m>:TYPE` on page 1670

Decode
Enables the decoding of the selected bus. The signal icon of the bus appears on the signal bar.

Remote command:
`BUS<m>[:STATe]` on page 1670

Data
Sets the source of the selected data line.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Alternatively, digital channels can be used if MSO option R&S RTO6-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

A math waveform can be used, for example, if you probe on CAN-High and CAN-Low using two single-ended probes, and the difference between high and low is calculated and displayed using a math waveform.

Remote command:
`BUS<m>:CAN:DATA:SOURce` on page 1795

Type
Selects the CAN-High or CAN-Low line. CAN uses both lines for differential signal transmission.

If you measure with a differential probe, connect the probe to both CAN-H and CAN-L lines, and select the data "Type" = *CAN-H*.

If you use a single-ended probe, connect the probe to either CAN_L or CAN_H, and select the data type accordingly.

Remote command:
`BUS<m>:CAN:TYPE` on page 1796

CAN-FD
Enables the CAN FD protocol configuration and displays additional CAN FD parameters.

The setting is available in CAN FD option.

Remote command:
`BUS<m>:CAN:FDATa:ENABle` on page 1798
`BUS<m>:CAN:FDATa:FRAMe<n>:STANdard?` on page 1809

Standard
Only available for CAN FD buses.

"Non-ISO"	Signals are decoded according to the the Bosch CAN FD protocol.
"ISO"	Signals are decoded according to the the ISO CAN FD protocol. This protocol has an additional stuff count field before the CRC sequence.

Remote command:
`BUS<m>:CAN:FDATa:PSTandard` on page 1796

Threshold
Sets the threshold value for digitization of the signal. If the signal value on the line is higher than the threshold, the signal state is high (1 or true for the boolean logic). Otherwise, the signal state is considered low (0 or false).

There are several ways to set the threshold:
- "Threshold"
  Enter the value directly in the field.
- "Autoset"
  Starts software algorithms for determining the signal threshold levels and bitrate.
- "Find thresholds"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Note: If the sources are digital channels, the same threshold values are used for the parallel and the serial buses. You can set the thresholds either in the parallel bus configuration or in the serial bus configuration.

Remote command:
`BUS<m>:CAN:DATA:THReshold` on page 1796
`BUS<m>:CAN:TECHnology` on page 1797
`BUS<m>:SETReflevels` on page 1671
`BUS<m>:FAUToset` on page 1671

Bit rate (CAN) / Arbitration rate (CAN FD)
For CAN buses, the "Bit rate" sets the number of transmitted bits per second.

For CAN FD buses, this parameter is called "Arbitration rate" and sets the bit rate of the arbitration phase.

The maximum value of this rate is 1 Mbit/s. The bit rate is uniform and fixed for a given CAN or CAN FD bus.

To select a bit rate from the list of predefined values, tap the button beside the field. To enter a specific value, open the keypad. The list of predefined values is also available in the keypad.

Remote command:
`BUS<m>:CAN:BITRate` on page 1797

Data rate
The setting is available in CAN FD option.

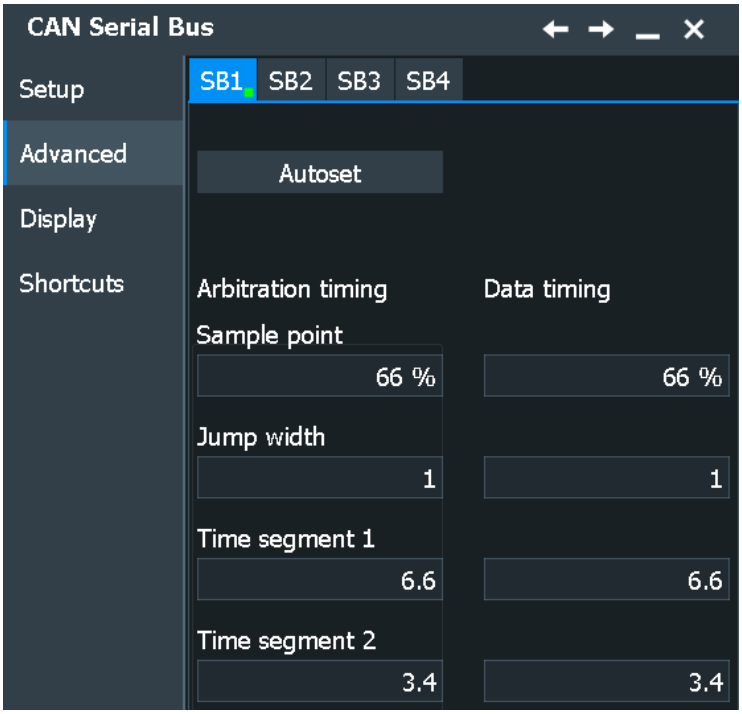
Sets the bit rate of the data phase. The data rate can be equal or higher than the arbitration rate; and it is uniform and fixed for a given CAN FD bus.

To select a data rate from the list of predefined values, tap the button beside the field. To enter a specific value, open the keypad. The list of predefined values is also available in the keypad.

Remote command:
`BUS<m>:CAN:FDATa:DBITrate` on page 1798

13.7.2.2 Advanced settings

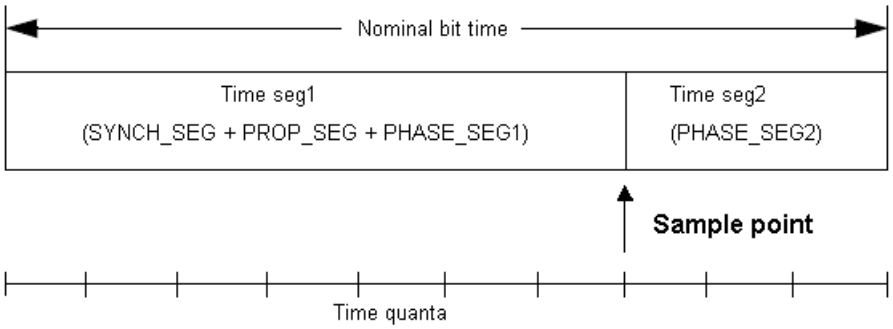
Access: [Bus] > "Advanced" tab.



The CAN bus interface uses an asynchronous transmission scheme. The standard specifies a set of rules to resynchronize the local clock of a CAN node to the message.

The sample point divides the nominal bit period into two distinct time segments. The length of the time segments is defined in time quanta according to network and node conditions during CAN development.

For CAN FD signals, you can define the sysnchronization settings separately for the arbitration phase and data phase.



To specify the bit timing, enter either "Time segment1" and "Time segment 2", or directly the "Sample point". Additionally, set the "Jump width".

Autoset
Starts software algorithms for determining the signal threshold levels and bitrate.

Remote command:
`BUS<m>:FAUToset` on page 1671

Time segment1, Time segment2
Set the number of time quanta before the sample point (Time segment 1) and after the sample point (Time segment 2).

The "Sample point" percentage value is adjusted accordingly.

Time segments 1 comprises the segments Synch_seg, Prop_seg, and Phase_seg1 which are specified in the CAN standard. Time seg2 matches Phase_seg2 from the standard.

The maximum sum of Time segment 1 and Time segment 2 is 24.

Remote command:
`BUS<m>:CAN:T1Segment` on page 1799
`BUS<m>:CAN:T2Segment` on page 1799
`BUS<m>:CAN:FDATa:T1Segment` on page 1799
`BUS<m>:CAN:FDATa:T2Segment` on page 1799

Sample point
Sets the position of the sample point within the bit in percent ot the nominal bit time.

The time quanta values "Time segment1, Time segment2" are adjusted accordingly.

Remote command:
`BUS<m>:CAN:SAMPlepoint` on page 1798
`BUS<m>:CAN:FDATa:SAMPlepoint` on page 1798

Jump width
Time segment1 may be lengthened or time segment2 may be shortened due to resynchronization. Resynchronization corrects the phase error of an edge caused by the drift of the oscillators. The jump width defines the maximum number of time quanta for phase correction. The maximum value of the jump width is 4, or *Time seg1 - Time seg2* if this difference is lower than 4.

Remote command:
`BUS<m>:CAN:JWIDth` on page 1800
`BUS<m>:CAN:FDATa:JWIDth` on page 1800

13.7.2.3 Display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

DBC Files for CAN / CAN FD

Industry-standard DBC files contain more information than PTT and CSV files and translate the abstract decode results to human language. For each frame, the frame ID and the symbolic name of the ID are given; the frames are also called messages in CAN. The data of a CAN message can consist of several "signals". The DBC file provides the label, unit, start bit, length and other indicators for each signal. For state-encoded signals, the meaning of the states is given.

In the demo example, the message "EngineData" has the decimal ID 2,166,573,756 and consists of 8 data bytes. These 8 bytes are defined as 6 signals. The first one,

"PetrolLevel", starts at bit #24, has a length of 8 bit, and the unit is liter. The signal "IdleRunning" is state-encoded. It has only one bit. The binary value 0 means "Running", and the binary value 1 means "Idle".

Example: CAN DBC file section

```
BO_ 2166573756 EngineData: 8 Engine
 SG_ PetrolLevel : 24|8@1+ (1,0) [0|255] "l" ...
 SG_ EngPower : 48|16@1+ (0.01,0) [0|350] "kW" ...
 SG_ EngForce : 32|10@1+ (1,0) [0|1000] "N" ...
 SG_ IdleRunning : 23|1@1+ (1,0) [0|1] "" ...
 SG_ EngTemp : 16|7@1+ (2,-50) [-50|150] "degC" ....
 SG_ EngSpeed : 0|13@1+ (1,0) [0|8000] "rpm" ...
....
 VAL_ 2166573756 IdleRunning 0 "Running" 1 "Idle" ;
```

PTT and CSV Files for CAN / CAN FD

Label list files are protocol-specific. A PTT label file for CAN protocols contains three values for each identifier:

- Identifier type, 11-bit or 29-bit long
- Identifier value
- Label, symbolic name of the identifier, specifying its function in the bus network.

Example: CAN PTT file

```
# ---------------------------------------------------------------------------
@FILE_VERSION  = 1.00
@PROTOCOL_NAME = can
# ---------------------------------------------------------------------------
# Labels for CAN protocol
#   Column order: Identifier type, Identifier value, Label
# ---------------------------------------------------------------------------
11,064h,Diag_Response
11,1E5h,EngineData
11,0A2h,Ignition_Info
11,1BCh,TP_Console
11,333h,ABSdata
11,313h,Door_Left
11,314h,Door_Right
29,01A54321h,Throttle
29,13A00FA2h,LightState
29,0630ABCDh,Engine_Status
29,03B1C002h,Airbag_Status
29,01234ABCh,NM_Gateway
# ---------------------------------------------------------------------------
```

Remote command:

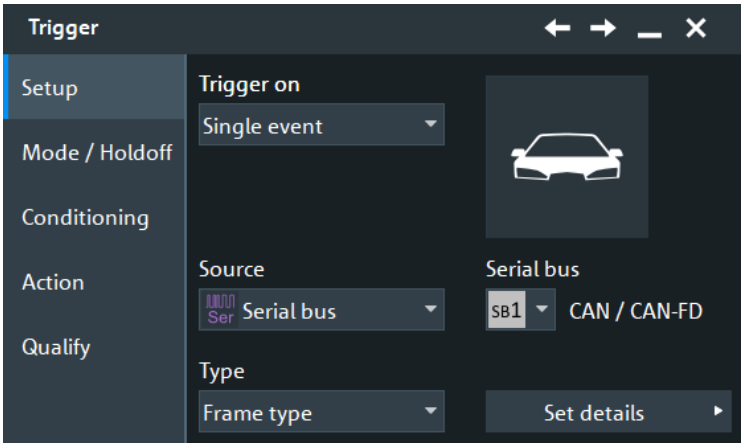
- `BUS<m>:CAN:FRAMe<n>:SYMBol?` on page 1811

13.7.3 CAN / CAN FD trigger

The R&S RTO6 can trigger on various events in a CAN or CAN FD frame. Trigger conditions include start of frame, frame ID, data pattern, or error conditions.

13.7.3.1 Trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Serial bus
Selects the serial bus to be triggered on. Make sure to select the correct bus before you enter the settings.

To trigger on a serial bus, the signals sources must be channel signals. If the data or clock source is a math or reference waveform, you cannot trigger on that bus.

Remote command:
`TRIGger<m>:SOURce:SBSelect` on page 1675

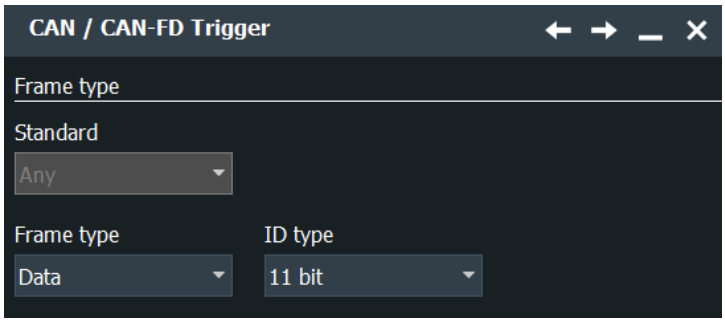
Protocol
Defines the protocol type of the selected serial bus.

Remote command:
`BUS<m>:TYPE` on page 1670

Trigger type
Selects the trigger type for CAN analysis.

"Start of frame" Triggers on the first edge of the dominant SOF bit (synchronization bit).

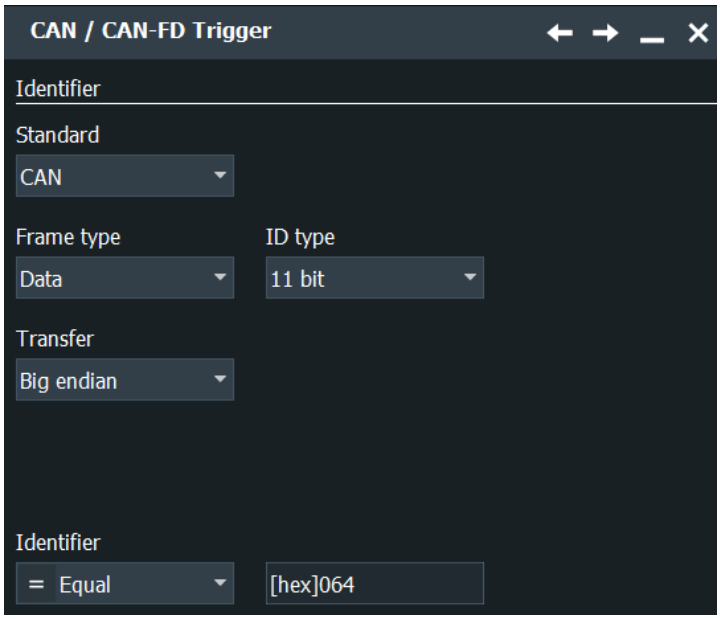
"Frame type"



Triggers on a specified frame type (data, remote, error, or overload). For data and remote frames, also the identifier format is considered.

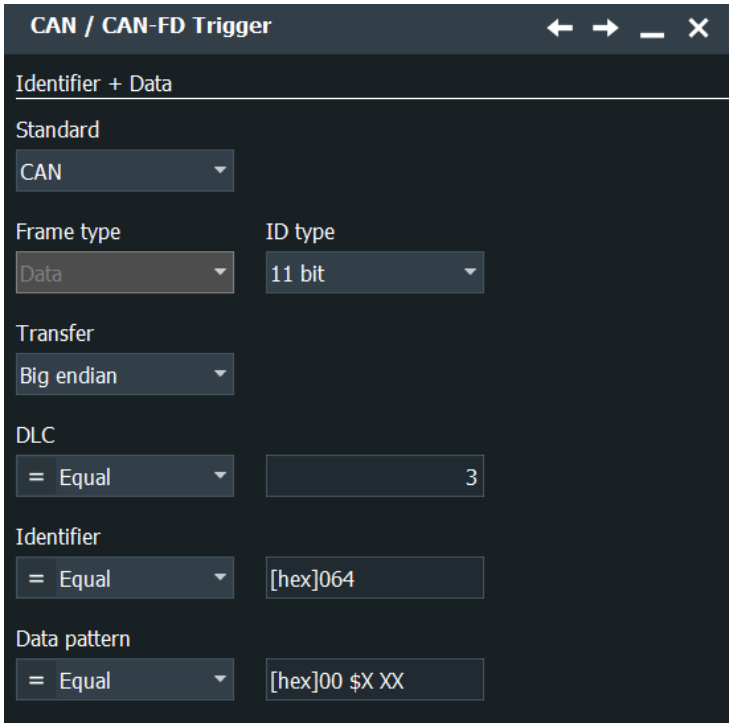
For details, see:
- "Frame type" on page 589
- "ID type" on page 589

"Identifier"



Sets the trigger to a specific message identifier or an identifier range. See "Identifier setup: Condition, Identifier min, Identifier max" on page 590.

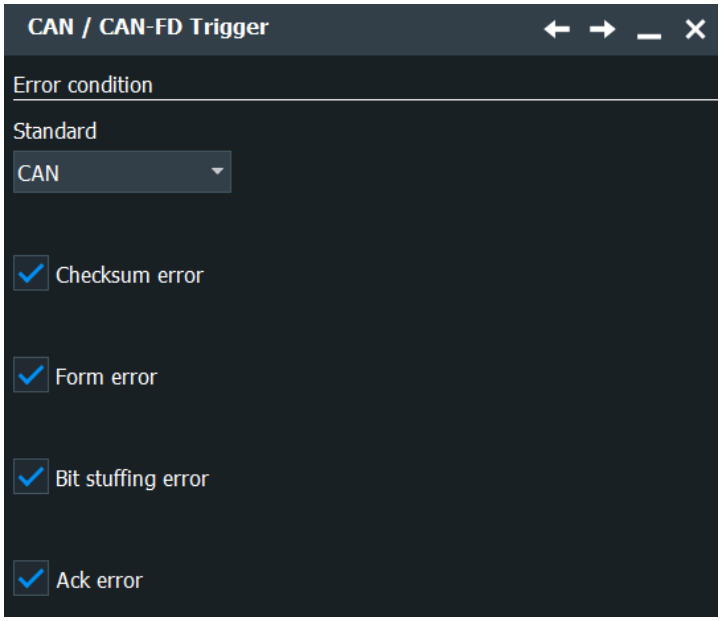
"Identifier + Data"



Sets the trigger to a combination of identifier and data condition. The instrument triggers at the end of the last byte of the specified data pattern.
The identifier conditions are the same as for the "Identifier" trigger type, see "Identifier setup: Condition, Identifier min, Identifier max" on page 590. Data conditions are set with "Data setup: DLC, NDB, Transfer, Condition, Data min, Data max" on page 591.

"Error condition"

Identifies various errors in the frame, see "Error conditions" on page 592.



"Symbolic" The "Symbolic" trigger type is available if a DBC label list file is loaded and applied. It allows you to trigger on a specific data message, or a signal and its value that appears inside the message, see Chapter 13.7.6.1, "Symbolic trigger", on page 600.

Remote command:
`TRIGger<m>:CAN:TYPE` on page 1801

Standard
Selects the CAN standard: "CAN", "CAN FD", or "Any".

The setting is available in CAN FD option.

Use "Any" to trigger on either CAN or CAN-FD frame. In this case, the trigger configuration provides all possible settings, for CAN as well as for CAN FD.

Remote command:
`TRIGger<m>:CAN:FDATa:STANdard` on page 1802

Frame type
CAN has four frame types which can be used as trigger condition.

For data and remote frames, the identifier format has to be set with ID type.

"Data" The data frame is the only frame for actual data transmission.

"Remote" Remote frames are only available in the CAN protocol.
The remote frame initiates the transmission of data by another node.
The frame format is the same as of data frames but without the data field.

"Error" When a node recognizes an error, it cancels transmission by sending an error frame.
The instrument triggers seven bit periods after the end of the error flag that is marked by a dominant-recessive edge.
The ID type is irrelevant for error frames.

"Overload" When a node needs a delay between data and/or remote frames, it sends and overload frame.
The instrument triggers seven bit periods after the end of the overload flag that is marked by a dominant-recessive edge.
The ID type is irrelevant for overload frames.

Remote command:
`TRIGger<m>:CAN:FTYPe` on page 1802

ID type
Selects the length of the identifier:

"11 bit" Identifier length of the CAN base frame format. The instrument triggers on the sample point of the IDE bit (identifier extension flag).

"29 bit" Identifier length of the CAN extended frame format. The instrument triggers on the sample point of the RTR bit.

"Any" The ID type and ID pattern are not relevant for the trigger condition. If the trigger type is "Identifier", the instrument triggers on any identifier in the specified frame type.
If the trigger type is "Identifier + Data", set the "ID type" to "Any" if you want to trigger only on data.

Remote command:
`TRIGger<m>:CAN:ITYPe` on page 1802

Identifier setup: Condition, Identifier min, Identifier max
The identifier setup consists mainly of the condition and one or two identifier patterns. Additionally, ID type and frame type may qualify the identifier.

The trigger point depends on the ID type.

"Frame type" Data frames and remote frames contain an identifier. Select the frame type to be triggered on, or select "Any" if the frame type is not relevant.
In CAN FD, only "Data" frames are available.

"ID type" See: "ID type" on page 589.

"Condition" Defines the operator to set a specific identifier ("Equal" or "Not equal") or an identifier range.

"Identifier min" In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.
The length of the bit patterns is restricted to the selected "ID type".

"Identifier max" The second identifier pattern is required to specify a range with conditions "In range" and "Out of range".

"FD bits" See: "FD bits" on page 590.

Remote command:
`TRIGger<m>:CAN:ICONdition` on page 1803
`TRIGger<m>:CAN:IMIN` on page 1803
`TRIGger<m>:CAN:IMAX` on page 1803

FD bits
For standard settings " CAN FD" and "Any", you can trigger on CAN FD-specific bits.

"FDF" The bit determines whether a frame is CAN or CAN-FD. It corresponds to the EDL bit (extended data length), which only exists in CAN FD format. If you do not know if the signal is CAN or CAN FD, you can use this bit to identify the format: FDF = 1 is CAN FD, and FDF = 0 is CAN. Set "X" if the format is not relevant.

"BRS" is the bit rate switch bit. Value 1 means that the bit rate switches from the "Arbitration rate" to the faster "Data rate".

"ESI" is the error state indicator. Set "X" if the bit is not relevant.

Remote command:
`TRIGger<m>:CAN:FDATa:FDF` on page 1803
`TRIGger<m>:CAN:FDATa:BRS` on page 1804
`TRIGger<m>:CAN:FDATa:ESI` on page 1804

Data setup: DLC, NDB, Transfer, Condition, Data min, Data max
The data setup consists of the transfer direction, the number of bytes, the condition, and one or two data patterns.

To trigger only on data, set the "ID type" of the identifier setup to "Any".

"Transfer" CAN only:

Sets the byte order (endianness) of the data transfer. With "Big endian", the data is analyzed and evaluated in the order of reception. With "Little endian", the instrument reads the complete data, reverses the byte order of the data, and compares it with the specified data word.

"DLC, NDB" "DLC" sets the Data Length Code, which defines the number of data bytes to be found.
"NDB" shows the number of data bytes that is set by the DLC. DLC and NDB are different in CAN FD for DLCs > 8.
CAN:

For Big Endian transfer direction, you can trigger on a number of bytes less than the data length of the frame, that means, on the first bytes that are transmitted. For Little Endian transfer direction, the exact number of data bytes in the frame must be set.
Example: The data word to be sent is *12 34 56*, and it is sent little endian by the LIN node. With Data length ≥ 2 and Transfer = Big endian, you trigger on the data of the first two bytes, that is *56 34*. With Data length = 3 and Transfer = Little endian, you trigger on the required data word *12 34 56*.
CAN FD:
The data field can have up to 64 bytes, the DLC is defined in the standard. For example, DLC = 9 defines that the data field has 12 bytes, and DLC = 15 sets a 64 byte data field.

"Condition" Sets the operator to set a specific data pattern ("Equal" or "Not equal") or a data range.

"Data min" Defines the data pattern. The pattern length is adjusted to the DLC setting (and vice versa). Enter the pattern MSB first and with big endian byte order.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:CAN:BORDer` on page 1805
`TRIGger<m>:CAN:DCONdition` on page 1804
`TRIGger<m>:CAN:DMIN` on page 1804
`TRIGger<m>:CAN:DMAX` on page 1805
`TRIGger<m>:CAN:DLCCondition` on page 1805
`TRIGger<m>:CAN:DLC` on page 1805
`TRIGger<m>:CAN:NDBYtes?` on page 1806

Data position

The data position sets the location in the data field where the instrument looks for the specified data pattern.

The setting is available in CAN FD option.

The position can be defined if the data field of the frame is longer than 8 bytes - if DLC ≥9.

"Position"	Sets the operator to define an exact position ("Equal") or a data range. Use "Any", if the data position is not relevant for the trigger condition.
"Data index (min)"	Defines the number of the first data byte at which the data pattern may start.
"Data index (max)"	Sets the number of the last byte at which the required data pattern may start if the "Position" operator is "In range".

Remote command:

`TRIGger<m>:CAN:FDATa:DPOPerator` on page 1806

`TRIGger<m>:CAN:FDATa:DPOSition` on page 1806

`TRIGger<m>:CAN:FDATa:DPTO` on page 1807

Error conditions

If a CAN detects an error, it transmits an error flag at the next bit. The R&S RTO6 detects errors in the message and triggers on these errors even if no CAN node sends an error flag.

- CRC error
  CAN uses the Cyclic Redundancy Check, which is a complex checksum calculation method. The transmitter calculates the CRC and sends the result in the CRC sequence. The receiver calculates the CRC in the same way. A CRC error occurs when the calculated result differs from the received value in the CRC sequence.
- Bit stuffing error
  The frame segments Start Of Frame, Arbitration Field, Control Field, Data Field and CRC Sequence are coded by the bit stuffing method. The transmitter automatically inserts a complementary bit into the bit stream when it detects five consecutive bits of identical value in the bit stream to be transmitted. A stuff error occurs when the 6th consecutive equal bit level in the mentioned fields is detected.
- Form error
  A form error occurs when a fixed-form bit field contains one or more illegal bits.
- Ack error
  An acknowledgement error occurs when the transmitter does not receive an acknowledgment - a dominant bit during the Ack Slot.
- Stuff count error
  A stuff count error occurs if th received stuff count value does not match the value calculated from the own stuff bit count. Only relevant for CAN FD signals in ISO standard.

Remote command:

`TRIGger<m>:CAN:CRCerror` on page 1807

`TRIGger<m>:CAN:BITSterror` on page 1807

`TRIGger<m>:CAN:FORMerror` on page 1807

TRIGger<m>:CAN:ACKerror on page 1807
TRIGger<m>:CAN:FDATa:SCERror on page 1808

13.7.3.2 Triggering on CAN FD data

The "Identifier + Data" trigger type supports triggering on data bytes of specific value at specific location in the data field of a frame. The "Data Pattern" field provides 8 data bytes to define the data pattern. For data fields longer than 8 byte, you can define the position where the specified pattern starts.

The following examples demonstrate how the data pattern and data position are defined.

To set up the trigger

1. Set the basic trigger events:
   a) Select the source: "Serial bus".
   b) Select the serial bus.
   c) Select the protocol: "CAN/CAN FD".
   d) Select the trigger type: "Identifier + Data".
   e) Select the standard:"CAN FD" or "Any".
2. In this example, the identifier does not matter. Set the "ID type = Any".
3. Define the data setup as described in the examples.

Example: Triggering on the second data byte

The CAN FD frame has 2 or more data bytes, where the value of the second data byte should be E7.

- Set "DLC ≥ 2".
- Set the data pattern: "= XX E7".

Example: Triggering on the seventh or later data byte

The CAN FD frame has 12 or more data bytes, where the value of the seventh or later data byte should be 17.

- Set "DLC ≥ 9".
- Set the data pattern: "= XX XX XX XX XX XX 17 XX".

Example: Triggering on a data byte at a given position

The CAN FD frame has 12 or more data bytes. the trigger is set at the 8th data byte starting at the 4th data byte or later, with data byte value = 17.

- Set "DLC ≥ 9".
- Set the data pattern: "= XX XX XX XX XX XX XX 17 ".
- Set the position of the data pattern: "In range", "4" to "12".

The instrument skips the first 3 data bytes and starts comparing the data pattern with the 4th data byte. So, the byte with value 17 can be found between the 11th and the 19th data byte.

13.7.4 CAN and CAN FD decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".

2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

The binary results of data bytes are displayed MSB first.

For CAN protocol, the endianness setting ("Transfer") is a trigger setting and not considered for decoding.

You can also load and apply industry-standard DBC files. The symbolic names from the file are applied to the display of the decoded data, see Chapter 13.7.6.2, "Symbolic decode waveform", on page 601.

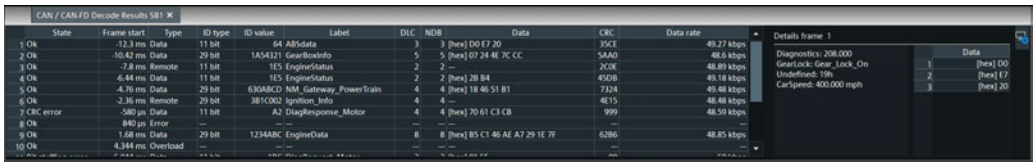


Table 13-10: Content of the Decode results table

Column	Description
State	Overall state of the frame. "Insuffcient waveform length" indicates that the frame is not completely contained in the acquisition. Change the horizontal scale, or move the reference point to the left to get a longer acquisition.
Frame start	Time of frame start
Type	Frame type: Data, Remote, Error, or Overload
ID type	11-bit standard format or 29-bit extended format
ID value (hex)	Identifier value, hexadecimal value
Standard	Frame format, CAN or CAN FD.
Label	Symbolic label name defined in the label list

Column	Description
DLC	Data length code, coded number of data bytes
NDB	Actual number of data bytes
Data	Value of the data frame. The data format is selected below the table. Remote frames do not transmit data, therefore "- - -" is displayed.
CRC (hex)	Value of the Cyclic Redundance Check (checksum), hexadecimal value
Data rate	Reason of a form error if a form error occurred

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands are described in Chapter 24.17.8.3, "Decode results", on page 1808.

13.7.5 Search on decoded CAN or CAN FD data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

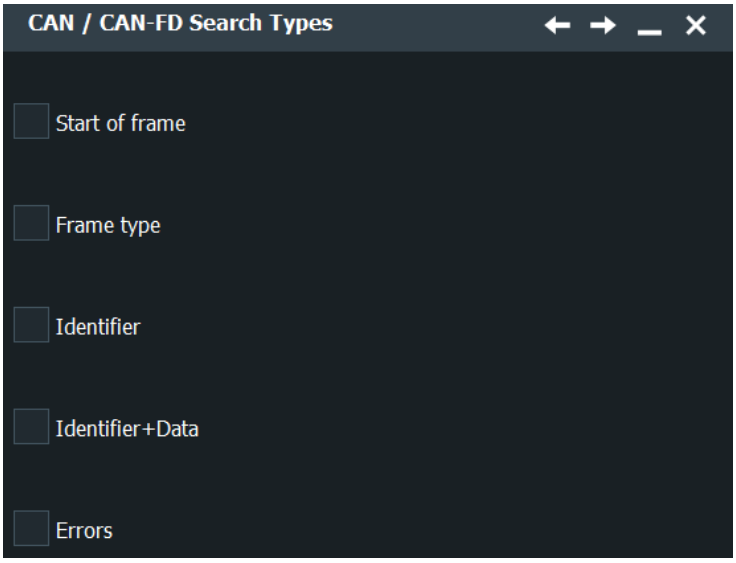
For general information on how to handle the search functionality, see Chapter 11, "Search functions", on page 419.

13.7.5.1 Search settings

Access: [Bus] > "Shortcuts" tab > "Setup search"

Search Criteria
Sets one criterion or an AND-combination of criteria to be searched for. If more than one criterion is selected, all criteria must be fulfilled by a frame for it to be shown in the search results.

If a DBC label list file is applied, an additional criterion "Symbolic" is provided, see Chapter 13.7.6.3, "Symbolic search", on page 602.



"Start of frame" Searches for the first edge of the dominant SOF bit (synchronization bit).

"Frame type" Searches for a specified frame type (data, remote, error, or overload). For data and remote frames, also the identifier format is considered.

For details, see:
- "Frame type" on page 597
- "ID type" on page 597

"Identifier" Searches for a specific message identifier or an identifier range. See "Identifier setup: Condition, Identifier min, Identifier max" on page 597.

"Identifier + Data"
Searches for a combination of identifier and data condition. The instrument triggers at the end of the last byte of the specified data pattern.
The identifier conditions are the same as for the "Identifier" search criteria, see "Identifier setup: Condition, Identifier min, Identifier max" on page 597. Data conditions are set with "Data setup: DLC, NDB, Condition, Data min, Data max" on page 598.

"Error condition"
Identifies various errors in the frame, see "Error conditions" on page 592.

"Symbolic" The "Symbolic" search criteria is available if a DBC label list file is loaded and applied. It allows you to search for specific data messages, or a signal and its value that appears inside the message. Symbolic search is an alternative to the other search criteria, you can either search for symbolic values or for an AND-combination of the other 5 criteria.
For details, see Chapter 13.7.6.3, "Symbolic search", on page 602.

Remote command:
SEARch:TRIGger:CAN[:SSOFrame] on page 1817
SEARch:TRIGger:CAN:SFTYpe on page 1818
SEARch:TRIGger:CAN:SFIDentifier on page 1818
SEARch:TRIGger:CAN:SIDData on page 1818
SEARch:TRIGger:CAN:SERRor on page 1818
SEARch:TRIGger:CAN:SSYMbolic on page 1834

Standard
Selects the CAN standard: "CAN", "CAN FD", or "Any".

The setting is available in CAN FD option.

Use "Any" to search for both CAN and CAN-FD frames. In this case, the search configuration provides all possible settings, for CAN as well as for CAN FD.

Remote command:
SEARch:TRIGger:CAN:FDATa:STANdard on page 1819

Frame type
Selects the frame type.

Remote frames are only available in the CAN protocol.

The frame types are the same as in the CAN trigger setup, see "Frame type" on page 589.

Remote command:
SEARch:TRIGger:CAN:FTYPe on page 1819

ID type
Selects the length of the identifier.

"11 bit"	Identifier length of the CAN base frame format. The instrument triggers on the sample point of the IDE bit.
"29 bit"	Identifier length of the CAN extended frame format. The instrument triggers on the sample point of the RTR bit.
"Any"	The ID type is not relevant. If the trigger type is "Identifier + Data", set the "ID type" to "Any" if you want to search only for data.

Remote command:
SEARch:TRIGger:CAN:ITYPe on page 1819

Identifier setup: Condition, Identifier min, Identifier max
The identifier setup consists mainly of the condition and one or two identifier patterns. Additionally, ID type and frame type may qualify the identifier.

The identifier setup settings are the same as in the CAN trigger setup, see "Identifier setup: Condition, Identifier min, Identifier max" on page 590.

FD bits: see "FD bits" on page 598.

Remote command:
SEARch:TRIGger:CAN:ICONdition on page 1819
SEARch:TRIGger:CAN:IMIN on page 1820
SEARch:TRIGger:CAN:IMAX on page 1820

FD bits
For standard settings " CAN FD" and "Any", you can search for CAN FD-specific bits.

For details, see "FD bits" on page 590.

The setting is available in CAN FD option.

Remote command:
`SEARch:TRIGger:CAN:FDATa[:FDF]` on page 1825
`SEARch:TRIGger:CAN:FDATa:BRS` on page 1825
`SEARch:TRIGger:CAN:FDATa:ESI` on page 1825

Data setup: DLC, NDB, Condition, Data min, Data max
The data setup consists of the number of bytes, the condition, and one or two data patterns.

The data setup settings are the same as in the CAN trigger setup, see "Data setup: DLC, NDB, Transfer, Condition, Data min, Data max" on page 591.

The data condition setting is also used for symbolic signal search, see Chapter 13.7.6.3, "Symbolic search", on page 602.

Remote command:
`SEARch:TRIGger:CAN:DCONdition` on page 1820
`SEARch:TRIGger:CAN:DMIN` on page 1821
`SEARch:TRIGger:CAN:DMAX` on page 1821
`SEARch:TRIGger:CAN:DLCCondition` on page 1821
`SEARch:TRIGger:CAN:DLC` on page 1822
`SEARch:RESult:CAN:FRAMe<m>:NDBYtes?` on page 1822

Data position
The data position sets the location in the data field where the instrument looks for the specified data pattern.

The setting is available in CAN FD option.

The position can be defined if the data field of the frame is longer than 8 bytes - if DLC ≥9.

For details, see "Data position" on page 592.

Remote command:
`SEARch:TRIGger:CAN:FDATa:DPOPerator` on page 1822
`SEARch:TRIGger:CAN:FDATa:DPOSition` on page 1823
`SEARch:TRIGger:CAN:FDATa:DPTO` on page 1823

Error Condition
Selects the error type to be searched for. You can select one or more error types as search condition.

The error types are the same as in the CAN trigger setup, see "Error conditions" on page 592.

Remote command:
`SEARch:TRIGger:CAN:CRCerror` on page 1824
`SEARch:TRIGger:CAN:BITSterror` on page 1824
`SEARch:TRIGger:CAN:FORMerror` on page 1824

`SEARch:TRIGger:CAN:ACKerror` on page 1823
`SEARch:TRIGger:CAN:FDATa:SCERror` on page 1824

13.7.5.2 Search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands are listed in Chapter 24.17.8.5, "Search results", on page 1825.

13.7.5.3 Searching CAN FD data

The "Identifier + Data" search supports the search for data bytes of specific value at a specific location in the data field of a frame. The "Data Pattern" field provides 8 data bytes to define the pattern. For data fields longer than 8 byte, you can define the position where the specified pattern starts.

The following example demonstrates how the data pattern and data position is defined.

To set up the search

1. Set the "Source", the signal to be searched: "SerBus". Select the bus that is configured for CAN FD.
2. Set the search criteria:
   a) Select the standard: "CAN FD".
   b) Select the search type: "Identifier + Data".
3. In this example, the identifier does not matter. Set the "ID type = Any".
4. Define the data setup as described in the example.

Example: Searching for a specific byte anywhere in the frame

The CAN FD frame has 8 or more data bytes, containing at least one data byte with value = CB anywhere in the data field.

- Set "DLC ≥ 8".
- Set the data pattern: "= CB".

13.7.6 Symbolic trigger, decode and search

The R&S RTO6 can read and apply industry-standard DBC files to the decoded signal and provides settings for symbolic triggering and symbolic search.

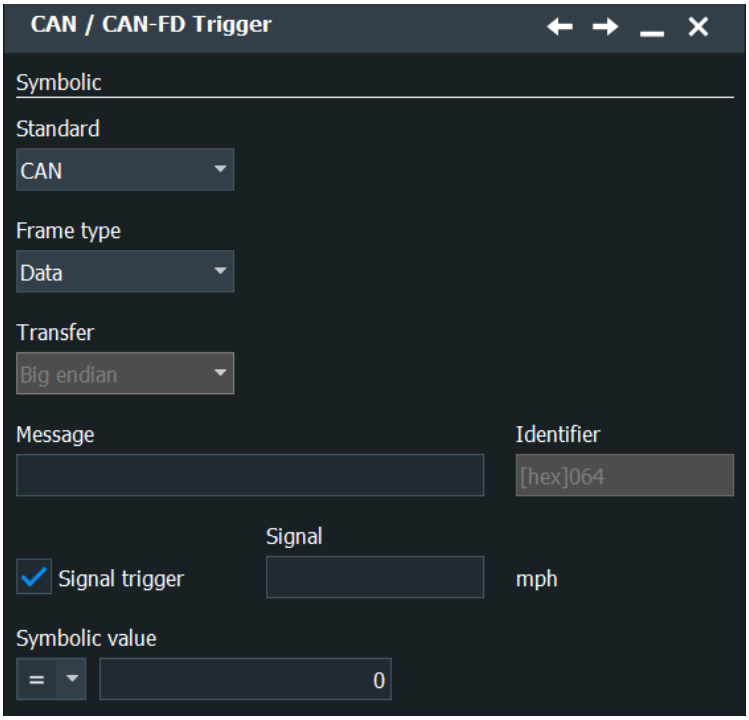
For a description of DBC files, see Chapter 13.7.2.3, "Display settings", on page 584.

13.7.6.1 Symbolic trigger

The "Symbolic" trigger type is available if a DBC label list file is loaded and applied. It allows you to trigger on a specific data message, or a signal and its value that appears inside the message.

Access:

1. Press [Bus].
2. In the "Setup" tab, set "Protocol = CAN/ CAN-FD".
3. In the "Shortcuts" tab, select "Setup trigger".
4. In the trigger dialog, set the "Type" = "Symbolic".



The "Frame type" is automatically set to "Data", and the "Identifier" is shown for information.

Specific settings for the symbolic trigger are:

Message....601
Signal trigger....601
Signal....601
Condition....601
Value, Value min....601
Value max....601

Message
Sets the message to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same message if symbolic trigger and symbolic search is used at the same time.

Remote command:
`TRIGger<m>:CAN:SYMBolic:MSGValue` on page 1832

Signal trigger
Enables the trigger on a specific signal value that is part of the selected message.

Remote command:
`TRIGger<m>:CAN:SYMBolic:TSIGnals` on page 1832

Signal
Sets the signal name to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same signal if symbolic trigger and symbolic search is used at the same time.

Remote command:
`TRIGger<m>:CAN:SYMBolic:SIGValue` on page 1832

Condition
Sets the operator to set a specific data pattern or symbolic value ("Equal" or "Not equal") or a data range.

Remote command:
`TRIGger<m>:CAN:DCONdition` on page 1804

Value, Value min
Defines the data pattern or selects a symbolic data value.

Remote command:
`TRIGger<m>:CAN:SYMBolic:DMIN` on page 1833
`TRIGger<m>:CAN:SYMBolic:SGEValue` on page 1833

Value max
The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:CAN:SYMBolic:DMAX` on page 1833

13.7.6.2 Symbolic decode waveform

If a DBC file is applied, the symbolic names from the file are applied to the display of the decoded data. The result table lists the signal values and units in the "Symbolic Data" column, and the comb display shows the signal names in addition to the signal values and units.

Remote command:

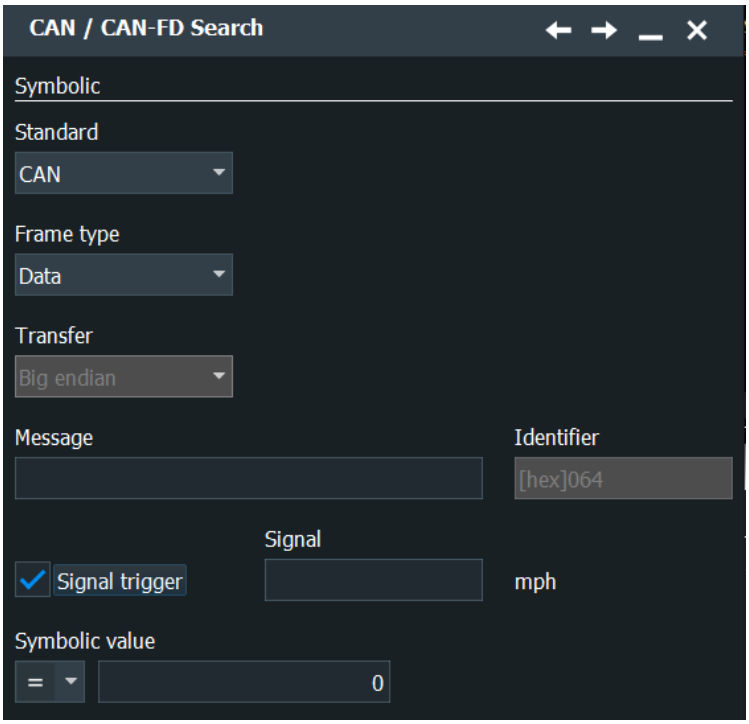
- `BUS<m>:CAN:FRAMe<n>:SDATa?` on page 1833
- `BUS<m>:CAN:FRAMe<n>:SDEXport?` on page 1815

13.7.6.3 Symbolic search

Access:

1. Press [Bus].
2. In the "Setup" tab, set "Protocol = CAN/ CAN-FD".
3. In the "Shortcuts" tab, select "Setup search".
4. In the search dialog, set the "Type" = "Symbolic".

If a DBC file is applied, the symbolic search for messages and signal, which are defined in the DBC file, is available. Symbolic search is an alternative to the other search criteria, you can either search for symbolic values or for an AND-combination of the other 5 criteria.



If symbolic search is active, the "Frame type" is automatically set to "Data", and the "Identifier" is shown for information.

Symbolic
Enables the symbolic search and disables all other search criteria.

Remote command:
`SEARch:TRIGger:CAN:SSYMbolic` on page 1834

Message
Sets the message to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same message if symbolic trigger and symbolic search is used at the same time.

Remote command:
`SEARch:TRIGger:CAN:SYMBolic:MSGValue` on page 1835

Search signals
Enables the search for a specific signal value that is part of the selected message.

Remote command:
`SEARch:TRIGger:CAN:SYMBolic:SSIGnals` on page 1835

Signal
Sets the signal name to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same signal if symbolic trigger and symbolic search is used at the same time.

Remote command:
`SEARch:TRIGger:CAN:SYMBolic:SIGValue` on page 1835

Condition
Sets the operator to set a specific data pattern or symbolic value ("Equal" or "Not equal") or a data range.

This condition is also used to search for data, see "Data setup: DLC, NDB, Condition, Data min, Data max" on page 598.

Remote command:
`SEARch:TRIGger:CAN:DCONdition` on page 1820

Value, Value (min)
Defines the data pattern or selects a symbolic data value.

Remote command:
`SEARch:TRIGger:CAN:SYMBolic:DMIN` on page 1836
`SEARch:TRIGger:CAN:SYMBolic:SGEValue` on page 1836

Value (max)
The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:CAN:SYMBolic:DMAX` on page 1836

13.7.6.4 Symbolic search results

If a DBC file is applied, you can search for symbolic messages and signals as described in Chapter 13.7.6.3, "Symbolic search", on page 602. As ususal, the search results

are shown in a table. You can enable the search zoom window to view the frame with the selected result in more detail.

The figure shows the result of a search for an "EngineData" message and the signal "IdleRunning = Running" inside the message. The result marker (blue line) is set to the start of the frame that fulfills the search condition. The search zoom window Search1 is active. It has been moved to the right until the "IdleRunning" bit with value 0 is visible in the zoom.

Remote command:

- `SEARch:RESult:CAN:FRAMe<m>:SDATa?` on page 1834

13.8 LIN (option R&S RTO6-K520)

The Local Interconnect Network (LIN) is a simple, low-cost bus system used within automotive network architectures. LIN is usually a subnetwork of a CAN bus. The primary purpose of LIN is the integration of uncritical sensors and actuators with low-bandwidth requirements. Common applications in a motor vehicle are the control of doors, windows, wing mirrors, and wipers.

13.8.1 The LIN protocol

This chapter provides an overview of protocol characteristics, frame format, identifiers and trigger possibilities. For detailed information, order the LIN specification on http://www.lin-subbus.org/ (free of charge).

LIN characteristics

Main characteristics of LIN are:

- Single-wire serial communications protocol, based on the UART byte-word interface
- Single primary, multiple secondaries - usually up to 12 nodes
- Primary-controlled communication: primary coordinates communication with the LIN schedule and sends identifier to the secondaries
- Synchronization mechanism for clock recovery by secondary nodes without crystal or ceramics resonator

The R&S RTO6 supports several versions of the LIN standard: v1.3, v2.0, v2.1 and the American SAE J2602.

Data transfer

Basic communication concept of LIN:

- Communication in an active LIN network is always initiated by the primary.
- The primary sends a message header including the synchronization break, the synchronization byte, and the message identifier.

- The identified node sends the message response: one to eight data bytes and one checksum byte.
- Header and response form the message frame.

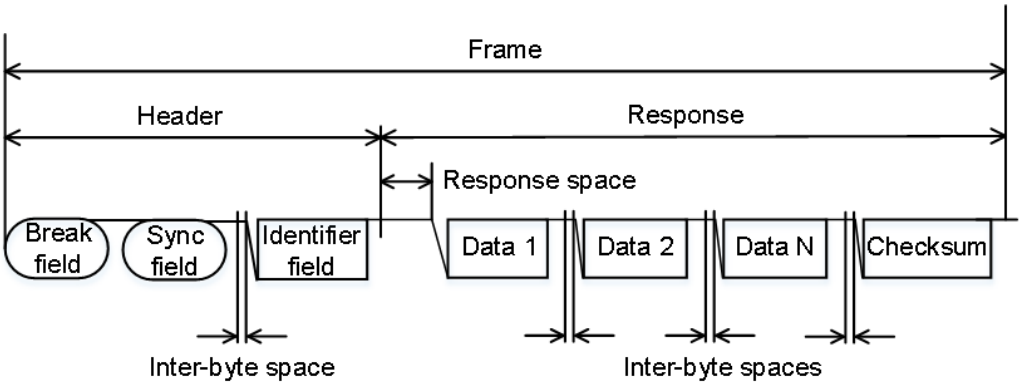


Figure 13-15: LIN frame with header and response

The data is transmitted in bytes using the UART byte-word interface without the parity bit. Each byte consists of a start bit, 8 bits and a stop bit.

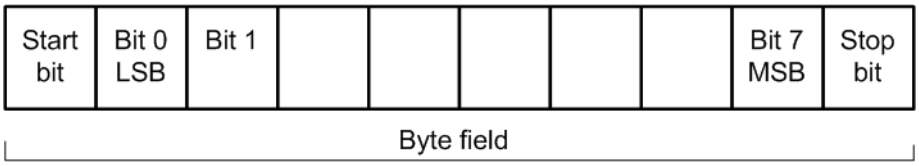


Figure 13-16: Structure of a byte field

Data bytes are transmitted LSB first.

The identifier byte consists of 6 bits for the frame identifier and two parity bits. This combination is known as protected identifier.

Trigger

The R&S RTO6 can trigger on various parts of LIN frames. The data line must be connected to an input channel, triggering on math and reference waveforms is not possible.

You can trigger on:

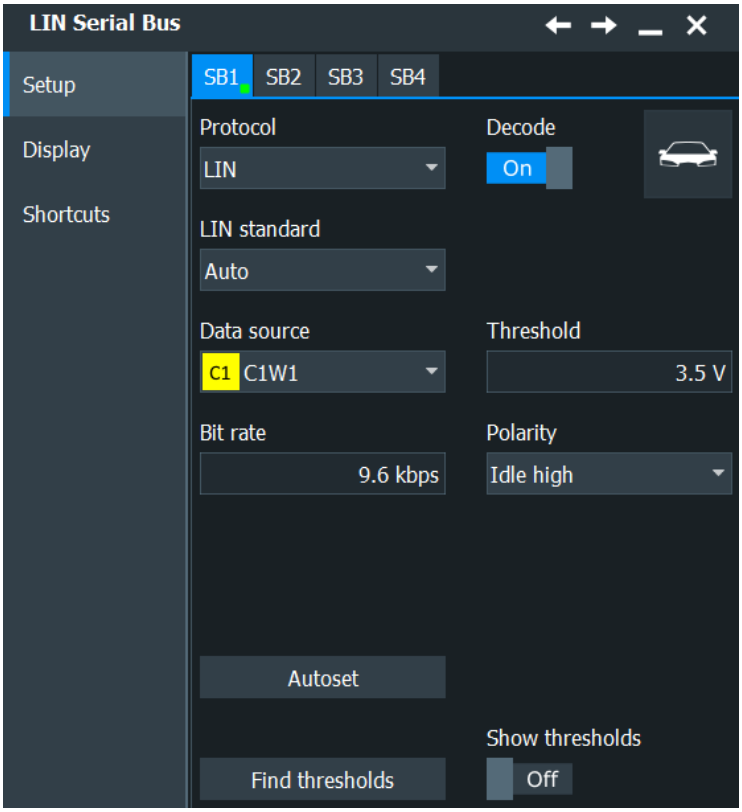
- Frame start (synchronization field)
- Specific identifier or identifier range
- Data pattern in the message
- Wake up signal
- Checksum error (error in data), parity error (error in identifier)

13.8.2 LIN configuration

13.8.2.1 LIN configuration settings

Access: [Bus] > "LIN" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

LIN standard
Selects the version of the LIN standard that is used in the DUT. The setting mainly defines the checksum version used during decoding.

The most common version is LIN 2.x. For mixed networks, or if the standard is unknown, set the LIN standard to "Auto".

Remote command:
`BUS<m>:LIN:STANdard` on page 1839

Data
Sets the source waveform of the data line.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Alternatively, digital channels can be used if MSO option R&S RTO6-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Remote command:
`BUS<m>:LIN:DATA:SOURce` on page 1837

Threshold
Sets the threshold value for digitization of the signal. If the signal value on the line is higher than the threshold, the signal state is high. Otherwise, the signal state is considered low if the signal value is below the threshold. The interpretation of HIGH and LOW is defined by the Polarity.

There are several ways to set the threshold:
- "Threshold"
  Enter the value directly in the field.
- "Autoset"
  Starts software algorithms for determining the signal threshold levels and bitrate.
- "Find thresholds"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Note: If the sources are digital channels, the same threshold values are used for the parallel and the serial buses. You can set the thresholds either in the parallel bus configuration or in the serial bus configuration.

Remote command:
`BUS<m>:LIN:DATA:THReshold` on page 1837
`BUS<m>:LIN:TECHnology` on page 1838
`BUS<m>:SETReflevels` on page 1671
`BUS<m>:FAUToset` on page 1671

Bit rate
Sets the number of transmitted bits per second. The maximum bit rate for LIN is 20 kbit/s.

To select a bit rate from list of predefined values, tap the icon beside the "Bit rate" field. To enter a specific value, open the keypad. The list of predefined values is also available in the keypad.

If the "LIN standard" is "J2602", the bit rate is 10.417 kbit/s and cannot be changed.

Remote command:
`BUS<m>:LIN:BITRate` on page 1838

Polarity
Defines the idle state of the bus. The idle state is the recessive state and corresponds to a logic 1.

Remote command:
`BUS<m>:LIN:POLarity` on page 1838

Show thresholds
If enabled, the threshold levels are displayed in the diagram.

Remote command:
`BUS<m>:THReshold` on page 1671

13.8.2.2 LIN display

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

LIN Specific Label Settings

Label lists are protocol-specific. Label lists for LIN are available in CSV and PTT format.

You can also apply an existing LIN description file.

A LIN label file contains two values for each identifier:

- Identifier value
- Symbolic name for the identifier

Example of a LIN PTT file

```
# ---------------------------------------------------------------------------
@FILE_VERSION = 1.0
@PROTOCOL_NAME = lin
# ---------------------------------------------------------------------------
# Labels for LIN protocol
#   Column order: Identifier, Label
# ---------------------------------------------------------------------------
# Labels for standard addresses
0x3F,Temperature
1Ch,Left brake
20h,Right brake
# Following ID is provided as integer
33,Mirror
0x37,Indoor lights
# Labels for reserved addresses
0x3C,Master_Request_Frame
0x3D,Slave_Response_Frame
# ---------------------------------------------------------------------------
```

The LIN description file (ldf) contains information about the assignment of the LIN frames to the nodes. The format and syntax are defined in the LIN specification.

A LIN ldf file contains the following sections:

- Header
- Node section
- Signal section
- Frame section

- Schedule table
- Signal encoding section
- Encoding to signal mapping

Example of a LIN LDF file

```
LIN_description_file;
LIN_protocol_version = "2.0";
LIN_language_version = "2.0";
LIN_speed = 19.2 kbps;

Nodes {
    Master: Master1, 10 ms, 0.1 ms;
    Slaves: Slave_Motor, Slave_Sensor;
}

Signals {
  Sensor1:1,0x0,Slave_Sensor, Master1;
  Temperature1:12,0,Slave_Sensor, Master1;
}

Frames {
  Sensor_Frame: {Error,30;
                 Temperature, 20;
                 Status, 0;}
  Master_Frame: {Error,15;
                 Speed, 10;}
}

Schedule_tables {
  Sensor_Frame delay 10.000 ms;
        Master_Frame delay 15.000 ms;
}

Signal_encoding_types {
  Speed {logical_value, 0, "Motor_off";
         logical_value, 1, "Speed1";
         logical_value, 2, "Speed2";
         logical_value, 3, "Speed3";
    }
}

Signal_representation {
  Speed1_encoding:Speed1
  Temperature1_encoding:Temperature1
}
```

Remote commands:

- `BUS<m>:LIN:FRAMe<n>:SDATa?` on page 1847
- `BUS<m>:LIN:FRAMe<n>:SDEXport?` on page 1847

13.8.2.3 Configuring LIN signals

For configuration, assign the lines to the input channels, and define the active states and the logical thresholds.

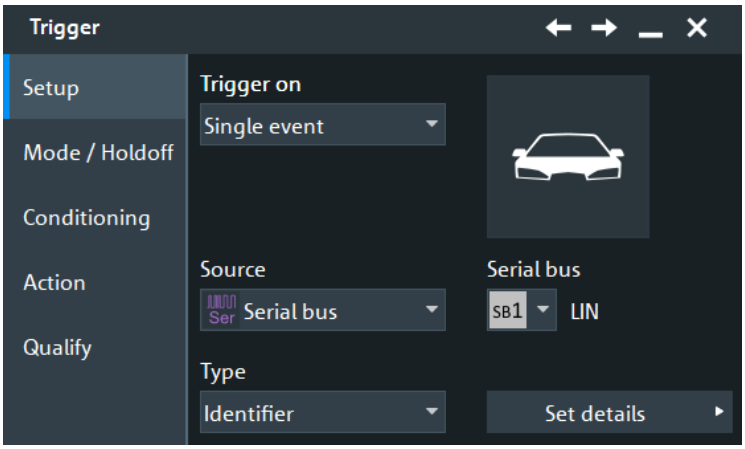
For details on configuration settings, see Chapter 13.8.2.1, "LIN configuration settings", on page 606.

1. Press the [Bus] key on the front panel.
2. At the left hand-side, select the vertical tab of the bus you want to set up.
3. Select the tab of the bus you want to set up, for example "SB1".
4. Tap the "Protocol" button and select the protocol: "LIN".
5. Set the "Data source".
6. Set the logical thresholds.
7. Set the "LIN standard", "Bit rate" and "Polarity" according to your signal.
8. Enable "Decode".

13.8.3 LIN trigger

13.8.3.1 LIN trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



Make sure that:

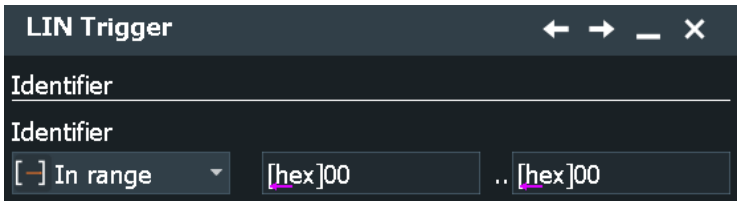
- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Trigger type

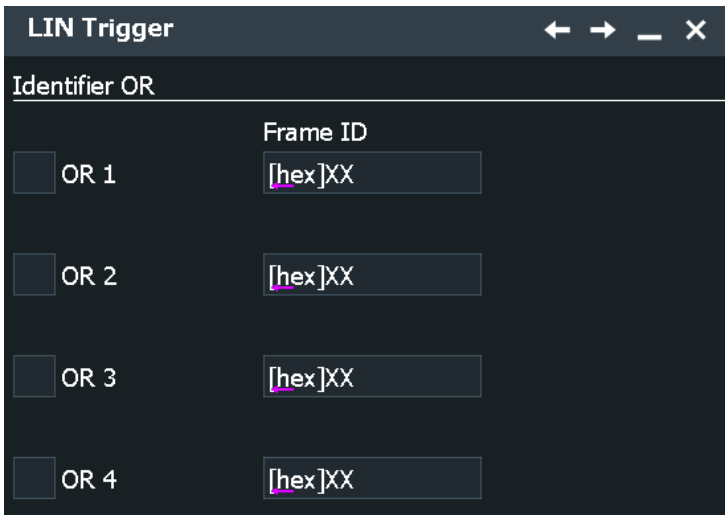
Selects the trigger type for LIN analysis.

"Start of frame (Sync)" Triggers on the stop bit of the sync field.

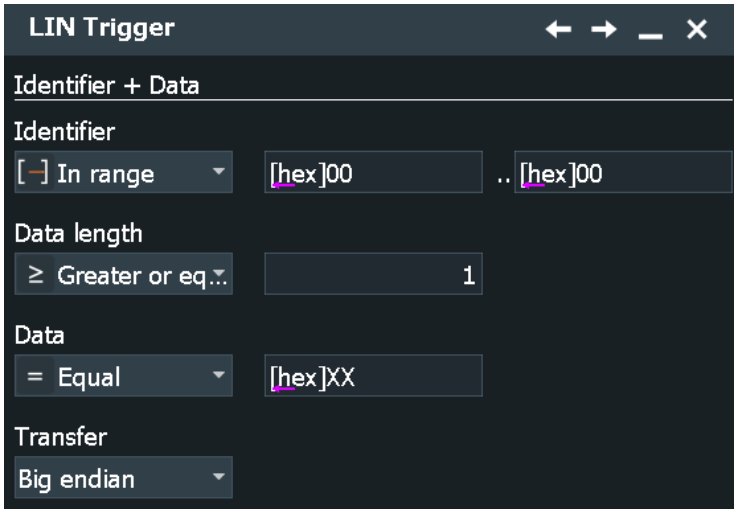
"Identifier" Sets the trigger to one specific identifier or an identifier range. Enter only the 6-bit identifier without parity bits, not the protected identifier. Description of trigger type specific settings: "Identifier setup: Condition, Frame ID min, Frame ID max" on page 612.



"Identifier OR" Sets the trigger to a combination of up to four identifiers. Description of trigger type specific settings: "Identifier OR setup: Monitor, Frame ID" on page 613



"Identifier + Data" — Sets the trigger to a combination of identifier and data condition. The instrument triggers at the end of the last byte of the specified data pattern.



The identifier conditions are the same as for the "Identifier" trigger type, see Identifier setup: Condition, Frame ID min, Frame ID max. Data conditions are set with Data setup: Data length, Transfer, Condition, Data min, Data max.

"Wakeup frame" — Triggers after a wakeup frame.

"Error condition" — Identifies various errors in the frame, see "Error conditions" on page 614.

Remote command:
`TRIGger<m>:LIN:TYPE` on page 1839

Identifier setup: Condition, Frame ID min, Frame ID max
The identifier setup consists of the condition and one or two identifier pattern.

"Condition" — Defines the operator to set a specific identifier ("Equal" or "Not equal") or an identifier range.

"Frame ID min / Frame ID" — Defines the bit pattern of the identifier. Enter only the 6-bit identifier without parity bits, not the protected identifier.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Frame ID max" — The second identifier pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:LIN:ICONdition` on page 1840
`TRIGger<m>:LIN:IMIN` on page 1841
`TRIGger<m>:LIN:IMAX` on page 1841

Identifier OR setup: Monitor, Frame ID
Sets the trigger to a combination of up to four identifiers. Enter the patterns in the "Frame ID" fields. In binary and hex format, characters 1, 0, and X (do not care) are allowed. For each identifier pattern to be triggered on, enable "Monitor".

Remote command:
`TRIGger<m>:LIN:IDOR<n>:ENABle` on page 1843
`TRIGger<m>:LIN:IDOR<n>[:VALue]` on page 1843

Data setup: Data length, Transfer, Condition, Data min, Data max
The data setup consists of the transfer direction, the number of bytes, the condition, and one or two data patterns.

"Transfer" Sets the byte order (endianness) of the data transfer. With "Big endian", the data is analyzed and evaluated in the order of reception. With "Little endian", the instrument reads the complete data, reverses the byte order of the data, and compares it with the specified data word.
According to the standard, LIN data is transmitted in little endian transfer order. The "Little endian" setting allows you to enter the required data word directly into "Data min", and the instrument triggers correctly.

"Data length" Sets the length of the bit pattern to be found, in bytes.

For Big Endian transfer direction, you can trigger on a number of bytes less than the data length of the frame, that means, on the first bytes that are transmitted. For Little Endian transfer direction, the exact number of data bytes in the frame must be set.
Example: The data word to be sent is *12 34 56*, and it is sent little endian by the LIN node. With Data length ≥ 2 and Transfer = Big endian, you trigger on the data of the first two bytes, that is *56 34*. With Data length = 3 and Transfer = Little endian, you trigger on the required data word *12 34 56*.

"Condition" Sets the operator to define a specific data pattern ("Equal" or "Not equal") or an data range.

"Data min" Defines the data pattern. The pattern length is adjusted to the data length setting (and vice versa), maximum is 8 bytes.
Enter the pattern MSB first and with big endian byte order, and set the correct "Transfer" direction. The data is compared byte by byte.
In binary format, use the following characters: 1; 0; or X (do not care).
The use of X is restricted to the operators "Equal" and "Not equal".

"Data max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:LIN:BORDer` on page 1842
`TRIGger<m>:LIN:DLECondition` on page 1842
`TRIGger<m>:LIN:DLENgth` on page 1842
`TRIGger<m>:LIN:DCONdition` on page 1841
`TRIGger<m>:LIN:DMIN` on page 1841
`TRIGger<m>:LIN:DMAX` on page 1841

Error conditions
Triggers if one or more of the following errors occur:

- Checksum error
  The checksum verifies the correct data transmission. It is the last byte of the frame response. The checksum includes not only the data but also the protected identifier (PID). To identify checksum errors caused by data, additional settings are required: Enter the bit pattern of the identifier ("Frame ID"), the number of data bytes ("Data length"), and select the used "LIN standard".
  See also: LIN standard.
- Identifier parity error
  Parity bits are the bits 6 and 7 of the identifier. They verify the correct transmission of the identifier.
- Sync error
  Synchronization error

Remote command:
`TRIGger<m>:LIN:CHKSerror` on page 1844
`TRIGger<m>:LIN:ERRPattern` on page 1844
`TRIGger<m>:LIN:CRCDatalen` on page 1844
`TRIGger<m>:LIN:STANdard` on page 1845
`TRIGger<m>:LIN:IPERror` on page 1843
`TRIGger<m>:LIN:SYERror` on page 1843

13.8.3.2 Triggering on LIN signals

Prerequisites: An LIN bus is configured, see Chapter 13.2.2.3, "Configuring I²C protocol", on page 499.

1. Press the [Bus] key and select the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to LIN.
5. Select the "Trigger type".
6. For more complex trigger types, enter the address and/or data conditions: address, identifier, frame ID.
   For details, see Chapter 13.8.3.1, "LIN trigger settings", on page 610.

13.8.4 LIN decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

The endianness setting ("Transfer") is a trigger setting and not considered for decoding. The binary results of data bytes are displayed MSB first.

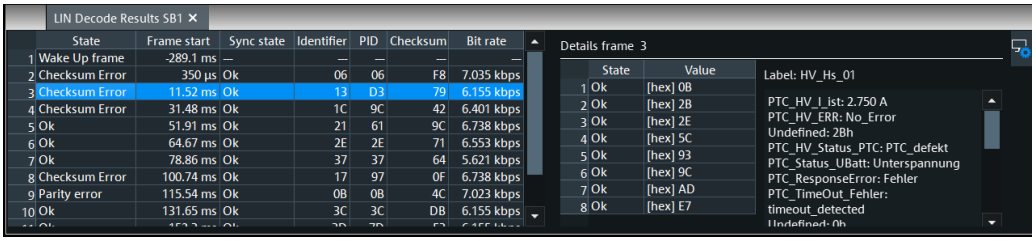


Table 13-11: Content of the Decode results table

Column	Description
State	Overall state of the frame.
Frame start	Time of frame start
Sync state	Result of synchronization
Identifier	Identifier value
Label	Symbolic label name defined in the label list
PID	Protected identifier
Values	Value of the data bytes. The data format is selected below the table.
Checksum	Checksum value
Bit rate	Value of the bit rate

Table 13-12: Content of the Details frame table

Column	Description
State	State of the field
Value	Value of the field
Label	Symbolic label name defined in the label list

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands are described in Chapter 24.17.9.3, "Decode results", on page 1845.

13.8.5 Search on decoded LIN data

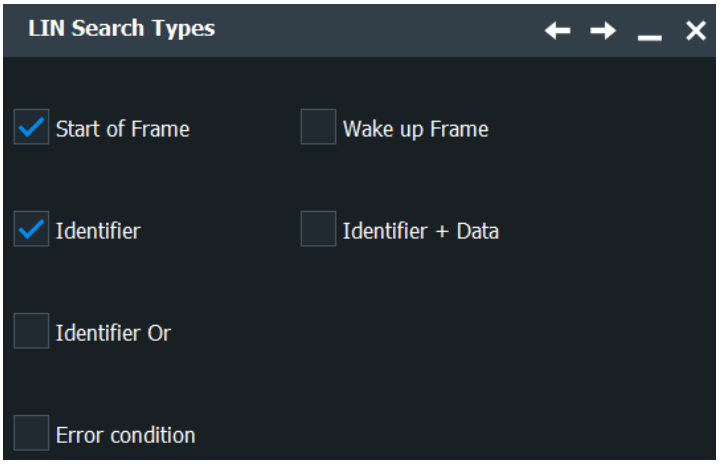
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 11, "Search functions", on page 419.

13.8.5.1 LIN search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"



Search types
Sets the type to be searched for.

"Start of frame (Sync)"	Searches for the stop bit of the sync field.
"Identifier"	Searches for one specific identifier or an identifier range. See "Identifier setup: Condition, Frame ID min, Frame ID max" on page 617
"Identifier OR"	Searches for a combination of up to four identifiers. See "Identifier OR setup: Monitor, Frame ID" on page 617

"Identifier + Data"	Searches for a combination of identifier and data condition. The instrument triggers at the end of the last byte of the specified data pattern. The identifier conditions are the same as for the "Identifier" trigger type, see "Identifier setup: Condition, Frame ID min, Frame ID max" on page 612. Data conditions are set with "Data setup: Data length, Transfer, Condition, Data min, Data max" on page 613.
"Wakeup frame"	Searches for wakeup frames.
"Error condition"	Identifies various errors in the frame, see "Error conditions" on page 614.

Remote command:
`SEARch:TRIGger:LIN:SSOFrame` on page 1853
`SEARch:TRIGger:LIN:SFIDentifier` on page 1853
`SEARch:TRIGger:LIN:IDENtifieror` on page 1853
`SEARch:TRIGger:LIN:SIDData` on page 1853
`SEARch:TRIGger:LIN:WUFRame` on page 1854
`SEARch:TRIGger:LIN:SERRor` on page 1854

Identifier setup: Condition, Frame ID min, Frame ID max
The identifier setup consists of the condition and one or two frame ID patterns.

The identifier setup settings are the same as in the LIN trigger setup, see "Identifier setup: Condition, Frame ID min, Frame ID max" on page 612.

Remote command:
`SEARch:TRIGger:LIN:ICONdition` on page 1854
`SEARch:TRIGger:LIN:IMIN` on page 1855
`SEARch:TRIGger:LIN:IMAX` on page 1855

Data setup: Condition, Data min, Data max, Data length, Transfer
The data setup consists of the transfer direction, the data length, the condition, and one or two data patterns.

The data setup settings are the same as in the LIN trigger setup, see "Data setup: Data length, Transfer, Condition, Data min, Data max" on page 613.

Remote command:
`SEARch:TRIGger:LIN:DCONdition` on page 1856
`SEARch:TRIGger:LIN:DMIN` on page 1856
`SEARch:TRIGger:LIN:DMAX` on page 1856
`SEARch:TRIGger:LIN:DLECondition` on page 1857
`SEARch:TRIGger:LIN:DLENgth` on page 1858
`SEARch:TRIGger:LIN:BORDer` on page 1857

Identifier OR setup: Monitor, Frame ID
The identifier OR setup consists of the monitor and frame ID.

The identifier OR setup settings are the same as in the LIN trigger setup, see "Identifier OR setup: Monitor, Frame ID" on page 613

Remote command:
`SEARch:TRIGger:LIN:IDOR<m>:ENABle` on page 1855
`SEARch:TRIGger:LIN:IDOR<m>[:VALue]` on page 1855

Error Condition
Selects the error type to be searched for. You can select one or more error types as search condition.

The error types are the same as in the LIN trigger setup, see "Error conditions" on page 614.

Remote command:
`SEARch:TRIGger:LIN:IPERror` on page 1858
`SEARch:TRIGger:LIN:SYERror` on page 1858
`SEARch:TRIGger:LIN:CHKSerror` on page 1858
`SEARch:TRIGger:LIN:ERRPattern` on page 1859
`SEARch:TRIGger:LIN:CRCDatalen` on page 1859
`SEARch:TRIGger:LIN:STANdard` on page 1859

13.8.5.2 LIN search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands are listed in Chapter 24.17.9.5, "LIN search results", on page 1859.

13.9 FlexRay (option R&S RTO6-K520)

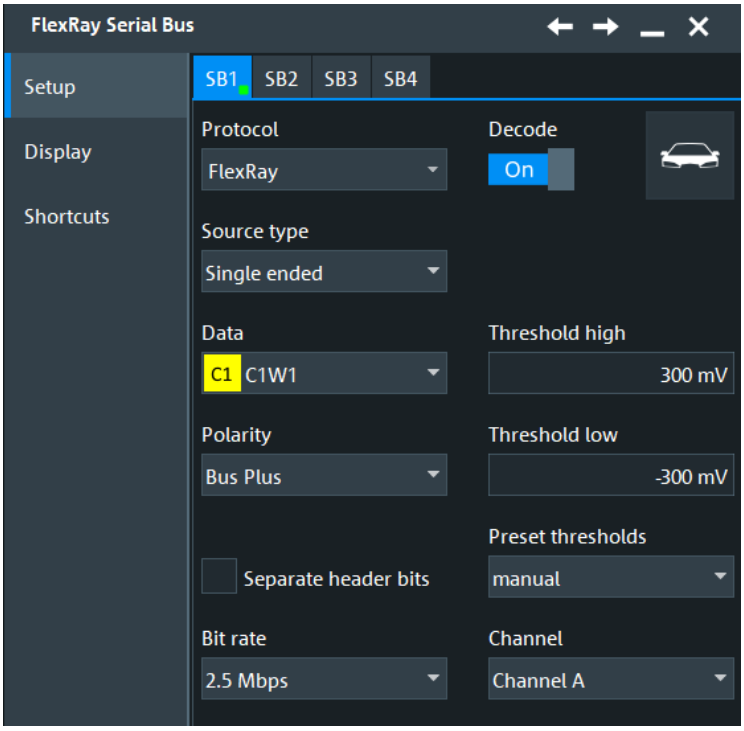
FlexRay is designed for use in safety-related distributed applications in the automotive industry. It is applied in real-time applications when higher data rates and reliable communication are required. In particular, FlexRay supports x-by-wire applications, for example, steer-by-wire or brake-by-wire.

13.9.1 FlexRay configuration

13.9.1.1 FlexRay configuration

Access: [Bus] > "Setup" tab > "Protocol" = *FlexRay*

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Source type
Sets the type of measurement. The instrument adjusts the thresholds to the selected source type.

"Single-ended" For measurements with single-ended probes, or single-ended voltage measurements with differential probes on the FlexRay bus. Two thresholds have to be defined as absolute voltage levels.

"Differential" For differential measurements on the FlexRay bus. This is the most common measurement. Two thresholds have to be defined as differential voltages.

"Logic" For measurements of logic signals, for example, of the logic signal inside the FlexRay node, between the communication controller and the bus driver. If MSO option R&S RTO6-B1 is installed, you can use digital input channels. It is possible to measure simultaneously on a data line and on the enable line. Each line requires its own threshold.

Remote command:
`BUS<m>:FLXRay:SRCType` on page 1865

Data
Sets the input channel of the bus signal, or of the data line for a "Logic" source type. Usually, the source is one of the analog channels. Reference and math waveforms are only available if the trigger source is one of the input channels but not the serial bus.

If the source type is "Logic", digital channels can be used (MSO option R&S RTO6-B1 is required). Digital and analog channels cannot be used at the same time in a bus.

Remote command:
`BUS<m>:FLXRay:SOURce<n>` on page 1866

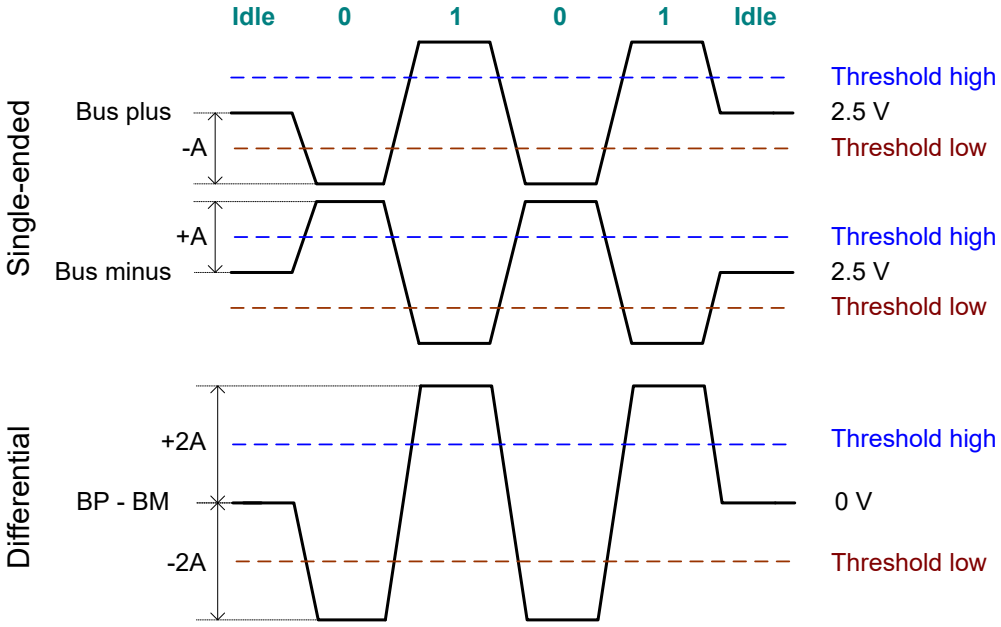
Enable
Sets the input channel of the enable line if there is a "Logic" source type. The enable line transfers the control signal of the bus guardian to the bus driver. None, or one of the analog channels can be used. Reference and math waveforms are only available if the trigger source is one of the input channels but not the serial bus.

Alternatively to analog channels, digital channels can be used if MSO option R&S RTO6-B1 is installed. Digital and analog channels cannot be used at the same time in a bus.

Remote command:
`BUS<m>:FLXRay:SOURce<n>` on page 1866

Thresholds
Threshold values are used for digitization of the signal.

For measurements on a FlexRay bus, two thresholds are required to distinguish the three possible states of the signal - high, low and idle. If the signal value on the line is higher than the upper threshold, the signal state is high. Otherwise, the signal state is considered low if the signal value is below the lower threshold. If the value is between the threshold, the signal is in idle state.



For measurements inside the FlexRay node (with "Source type" = "Logic"), each line requires its threshold level.

There are two ways to set the thresholds: selection of a predefined value, or direct entry of a value.

- "Preset thresholds"

Selects default threshold voltages from a list. The predefined values depend on the selected source type. The value is set to "Manual" if at least one threshold was entered directly.

- "Threshold high" and "Threshold low"
  Upper and lower levels for single-ended or differential source types. You can enter the values directly in the fields.
- "Threshold data" and "Threshold enable"
  Levels for data and enable line if there is logic source type. You can enter the values directly in the fields.

Note: If the sources are digital channels, the same threshold values are used for the parallel and the serial buses. You can set the thresholds either in the parallel bus configuration or in the serial bus configuration.

Remote command:
`BUS<m>:FLXRay:PRSingle` on page 1867
`BUS<m>:FLXRay:PRDiff` on page 1868
`BUS<m>:FLXRay:PRLogic` on page 1868
`BUS<m>:FLXRay:THReshold<n>` on page 1867
`BUS<m>:FLXRay:THData` on page 1867
`BUS<m>:FLXRay:THENable` on page 1867
`BUS<m>:SETReflevels` on page 1671

Polarity
Selects the wire on which the bus signal is measured if there is "Single-ended" measurement: "Bus plus" or "Bus minus". The setting affects the digitization of the signal.

Remote command:
`BUS<m>:FLXRay:POLarity` on page 1869

Bit rate
Selects the number of transmitted bits per second from a list.

Remote command:
`BUS<m>:FLXRay:BITRate` on page 1869

Channel
Selects the FlexRay channel on which the signal is measured, either channel A or channel B. The setting is considered in the calculation of the frame CRC.

Remote command:
`BUS<m>:FLXRay:CHTYpe` on page 1869

Separate header bits
The setting affects the decoding and its display. If enabled, the leading five indicator bits of the header are decoded as five single bits. Otherwise, the indicator bits are shown as one word with word length 5 bits.

Remote command:
`BUS<m>:FLXRay:SEHB` on page 1869

13.9.1.2 Display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

FlexRay Specific Label Settings

Label lists are protocol-specific. A FlexRay label file contains four values for each identifier:

- "ID / Addr": number of the slot in which the frame is transmitted
- "Base cycle" and "Repetition": define the cycle indexes for which the identifier applies. Base cycle defines the first applied cycle.
  There are 64 cycles in a FlexRay communication. The same identifier can be shared by different devices, and each device uses the identifier at different cycles. For example:
  0x0AB,0,2,Ignition_Info: uses cycles 0,2,4,6,...,62
  0x0AB,1,2,GearBoxInfo: uses cycles 1,3,5,7,...,63
- "Symbolic label": symbolic name of the identifier, specifying the device function.

Example: FlexRay PTT file

```
# ------------------------------------------------------------------------------
@FILE_VERSION = 1.0
@PROTOCOL_NAME = flexray
# ------------------------------------------------------------------------------
# Labels for FlexRay protocol
#   Column order: Identifier, Base cycle, Cycle repetition, Label
# ------------------------------------------------------------------------------
# ----Definition----
0x01D,0,2,DriveTrain
0x03D,3,3,EngineData
0x0AB,0,2,Ignition_Info
0x0AB,1,2,GearBoxInfo
0x1C4,1,2,ABSdata
0x1F5,0,1,BrakeControl
0x200,0,1,Engine_Status
0x2BA,0,1,Airbag_Status
0x2D9,0,1,TP_Console
0x340,0,2,CAN_Gateway
0x38B,55,1,MOST_Gateway
0x3EA,0,1,PressureInfo
# ------------------------------------------------------------------------------
```

13.9.1.3 Configuring FlexRay protocol

The configuration of the FlexRay is simple - assign the two lines to input channels, and set the thresholds.

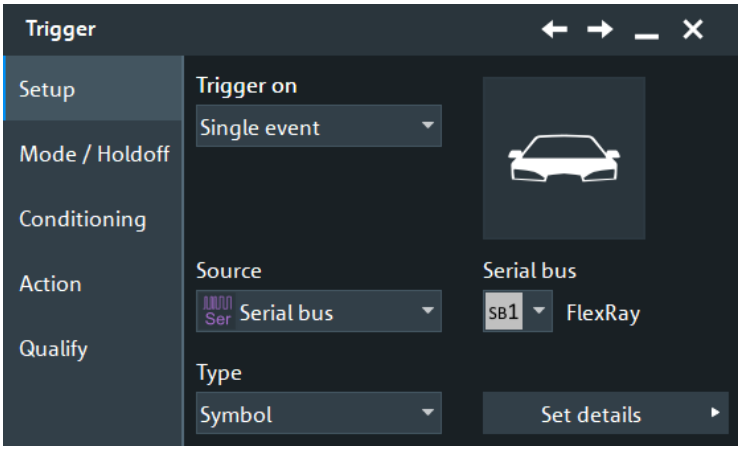
For details on configuration settings, see Chapter 13.9.1.1, "FlexRay configuration", on page 618.

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. Select the "Setup" tab.
4. Tap the "Protocol" button and select the protocol: "FlexRay".
5. Select the "Source type".
6. Select the channels for the sources.
7. Adjust other settings, if necessary.
8. Set the logical thresholds.
9. Enable "Decode".

13.9.2 FlexRay trigger

13.9.2.1 FlexRay trigger

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



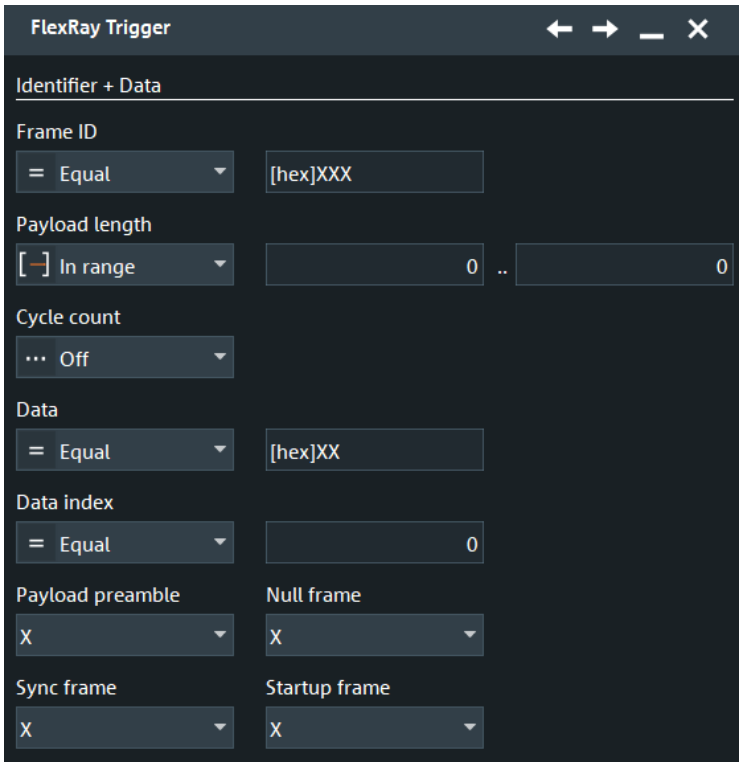
Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Trigger type
Selects the trigger type for FlexRay analysis.

"Start of frame" Triggers on the first rising edge after the transmission start sequence (TSS).

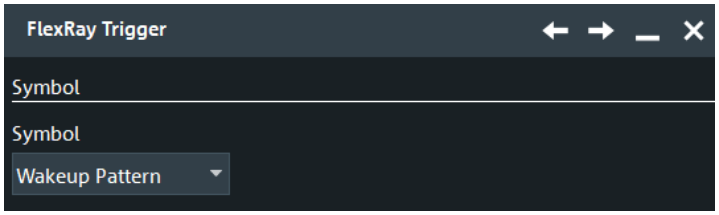
"Identifier +data"



Triggers on the decoded frame content, on header and payload data:

- Indicator bits, see "Indicator bits" on page 625
- Frame identifier, see "Frame ID (min/max)" on page 626
- Payload length, see "Payload length (min/max)" on page 626
- Cycle count, see "Cycle count (min, max), Step" on page 626
- Data position, see "Position, Index (min, max) - Data setup" on page 627
- Data bit pattern, see "Condition, Data (min, max) - Data setup" on page 627

"Symbol" Triggers on a symbol or wakeup pattern, see "Symbol" on page 628.

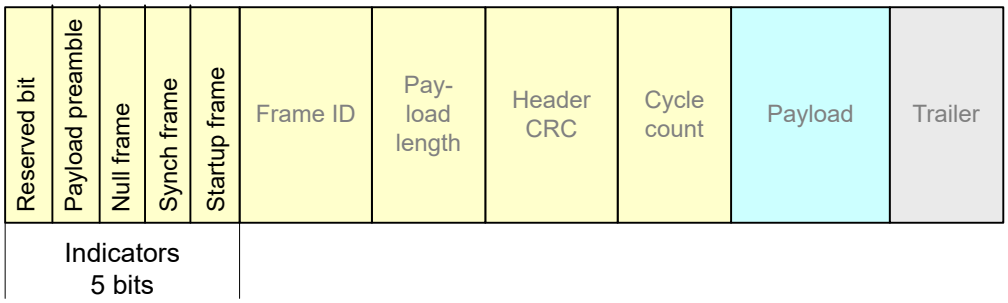


"Error condition" Triggers on one or more errors that are detected in the decoded data, see "Error conditions" on page 628.

Remote command:
`TRIGger<m>:FLXRay:TYPE` on page 1871

Indicator bits
Triggers on one or more indicator bits at the beginning of the header segment. Each bit can be set to 0, 1, or X (do not care).

Trigger type: "Identifier + data"



"Payload preamble" Indicates a Network Management Vector in the payload segment. The NMV allows the host processor to send data directly, without processing by the communication controller.

"Null frame" Indicates a frame without usable data.

"Sync frame" Indicates that the frame is used for synchronization of the FlexRay system. Only sync nodes can send this frame type.

"Startup frame" Indicates a startup frame used for startup of the network. Only specific start nodes can send this frame type.

Remote command:
`TRIGger<m>:FLXRay:PLPReamble` on page 1872
`TRIGger<m>:FLXRay:NUFRame` on page 1872

TRIGger<m>:FLXRay:SYFRame on page 1872
TRIGger<m>:FLXRay:STFRame on page 1872

Frame ID (min/max)
The frame ID contains the number of the slot in which the frame is transmitted. Each frame ID occurs only once during a FlexRay cycle.

Indica-tors	Frame ID	Payload length	Header CRC	Cycle count	Payload	Trailer
5 bits	11 bits	7 bits	11 bits	6 bits		

To trigger on a frame ID, you have to define a condition and one or two identifier patterns. The second identifier pattern is required to specify a range with conditions "In range" and "Out of range". In binary format, use the following characters: 1; 0; or X (any bit). The use of X is restricted to the conditions "Equal" and "Not equal". If the identifier is not relevant for the trigger setup, set it to "Off".

The maximum length of the pattern is 11 bit.

The bit pattern editor helps you to enter the pattern in any format, see Chapter 13.1.7, "Bit pattern editor", on page 493.

Trigger type: "Identifier + data"

Remote command:
TRIGger<m>:FLXRay:FCONdition on page 1872
TRIGger<m>:FLXRay:FMIN on page 1873
TRIGger<m>:FLXRay:FMAX on page 1873

Payload length (min/max)
The payload length contains the number of words transmitted in the payload segment. Information is transmitted in 2-byte words, so the number of data bytes in the payload segment is twice the payload length.

Indica-tors	Frame ID	Payload length	Header CRC	Cycle count	Payload	Trailer
5 bits	11 bits	7 bits	11 bits	6 bits		

To trigger on the payload length, you have to define a condition and one or two numbers of words. The second number is required to specify a range with conditions "In range" and "Out of range". If the payload length is not relevant for the trigger setup, set it to "Off".

Trigger type: "Identifier + data"

Remote command:
TRIGger<m>:FLXRay:PCONdition on page 1873
TRIGger<m>:FLXRay:PMIN on page 1874
TRIGger<m>:FLXRay:PMAX on page 1874

Cycle count (min, max), Step
The cycle count contains the number of the current FlexRay cycle.

Indica-tors	Frame ID	Payload length	Header CRC	Cycle count	Payload	Trailer
5 bits	11 bits	7 bits	11 bits	6 bits		

To trigger on the cycle count, you have to define a condition and one or two numbers. If the condition is a range ("In range" or "Out of range"), a second number "Cycle count max" is required.

Also, you can define a "Step" to trigger on each n-th cycle inside the given range. This allows for specific triggering if slot multiplexing is used.

If the cycle count is not relevant for the trigger setup, set it to "Off".

Trigger type: "Identifier + data"

Remote command:
`TRIGger<m>:FLXRay:CENable` on page 1874
`TRIGger<m>:FLXRay:CMIN` on page 1874
`TRIGger<m>:FLXRay:CMAX` on page 1875
`TRIGger<m>:FLXRay:CSTep` on page 1875

Position, Index (min, max) - Data setup
Sets the position of the first byte of data bit pattern within the payload segment. You can define an exact position, or a position range.

Trigger type: "Identifier + data"

"Position" Operator for the data position. Select "Off", if the position of the required pattern is not relevant for the trigger condition.

"Index" Sets the number of data bytes to be skipped after start of the payload segment if "Position" is "Equal" or "Greater or equal". The index 0 is associated with the first data byte.

"Index min, Index max"
If the "Position" operator defines a range, the indexes of the first and the last byte are defined between which the required bit pattern may start.

Remote command:
`TRIGger<m>:FLXRay:DPOPerator` on page 1875
`TRIGger<m>:FLXRay:DPOSition` on page 1875
`TRIGger<m>:FLXRay:DPTO` on page 1876

Condition, Data (min, max) - Data setup
Specifies the data bit pattern to be found in the payload segment. The starting point of the pattern is defined by "Position, Index (min, max) - Data setup" on page 627. The pattern comparison is byte-aligned, and the instrument triggers at the end of a byte.

"Condition" Sets the operator to set a specific data pattern ("Equal" or "Not equal") or a data range. Select "Off", if the data pattern is not relevant for the trigger condition.

"Data (min/max)" Enter the bytes in msb first bit order. The maximum pattern length is 8 bytes.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

Remote command:
`TRIGger<m>:FLXRay:DCONdition` on page 1876
`TRIGger<m>:FLXRay:DMIN` on page 1876
`TRIGger<m>:FLXRay:DMAX` on page 1876

Symbol
Triggers on a symbol or on a wakeup pattern.

Trigger type: "Symbol"

"CAS/MTS" Collision Avoidance Symbol / Media access Test Symbol. These symbols are identical and can be sent in the optional symbol window at the end of a communication cycle. They are used to avoid collisions during the system start.

"Wakeup Pattern" The wakeup pattern is sent to activate the nodes of the system.

Remote command:
`TRIGger<m>:FLXRay:SYMBol` on page 1877

Error conditions
Triggers on one or more errors in the frame.

Trigger type: "Error conditions"

"FSS" Error in a Frame Start Sequence. FSS follows the Transmission Start Sequence TSS at the beginning of each frame.

"BSS" Error in a Byte Start Sequence. The BSS is transmitted before each byte.

"FES" Error in Frame End Sequence. FES indicates the end of each frame.

"Header CRC" Error in a cyclic redundancy check code of the header data which covers mainly frame ID and payload length.

"Payload CRC" Error in a cyclic redundancy check code of the complete frame.

Remote command:
`TRIGger<m>:FLXRay:FSSerror` on page 1877
`TRIGger<m>:FLXRay:BSSerror` on page 1877
`TRIGger<m>:FLXRay:FESerror` on page 1877
`TRIGger<m>:FLXRay:HCRCerror` on page 1878
`TRIGger<m>:FLXRay:PCRCerror` on page 1878

13.9.2.2 Triggering on FlexRay signals

Prerequisites: An FlexRay bus is configured, see .

1. Press the [Bus] key and selet the "Shortcuts" tab.

2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to FlexRay.
5. Select the "Trigger type".
6. For more complex trigger types, enter the address and/or data conditions: address, acknowledge bits, R/W bit, and data pattern.
   For details, see Chapter 13.9.2.1, "FlexRay trigger", on page 623.

13.9.3 FlexRay decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

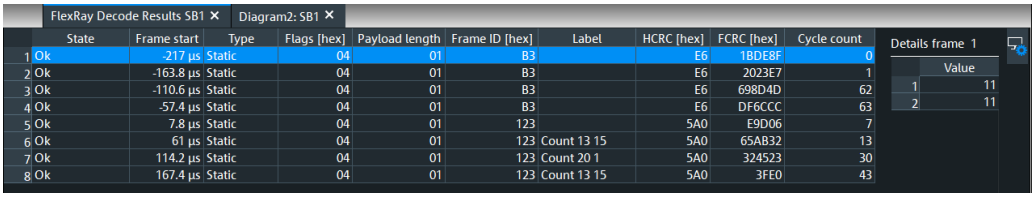
The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.



	State	Frame start	Type	Flags [hex]	Payload length	Frame ID [hex]	Label	HCRC [hex]	FCRC [hex]	Cycle count
1	Ok	-217 µs	Static	04	01	B3		E6	1BDE8F	0
2	Ok	-163.8 µs	Static	04	01	B3		E6	2023E7	1
3	Ok	-110.6 µs	Static	04	01	B3		E6	698D4D	62
4	Ok	-57.4 µs	Static	04	01	B3		E6	DF6CCC	63
5	Ok	7.8 µs	Static	04	01	123		5A0	E9D06	7
6	Ok	61 µs	Static	04	01	123	Count 13 15	5A0	65AB32	13
7	Ok	114.2 µs	Static	04	01	123	Count 20 1	5A0	324523	30
8	Ok	167.4 µs	Static	04	01	123	Count 13 15	5A0	3FE0	43

Table 13-13: Content of the Decode results table

Column	Description
State	Overall state of the frame. "Insuffcient waveform length" indicates that the frame is not completely contained in the acquisition. Change the horizontal scale, or move the reference point to the left to get a longer acquisition.
Frame start	Time of frame start

Column	Description
Type	Frame type: Frame of the static segment, frame of the dynamic segment, wakeup frame, symbol in the frame
Flags	State of indicator bits
Payload length	Number of data words in the payload segment.
Frame ID	Value of the frame ID (slot number)
Label	Symbolic label name defined in the label list
HCRC	Value of the header CRC
FCRC	Value of the frame CRC
Cycle count	Number of the current FlexRay cycle
Values	Value of the data bytes. The data format is selected below the table. Wakeup and symbol frames do not transmit data, therefore "- - -" is displayed.

Remote commands

Remote commands are described in Chapter 24.17.10.3, "Decode results", on page 1878.

13.9.4 Search on decoded FlexRay data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

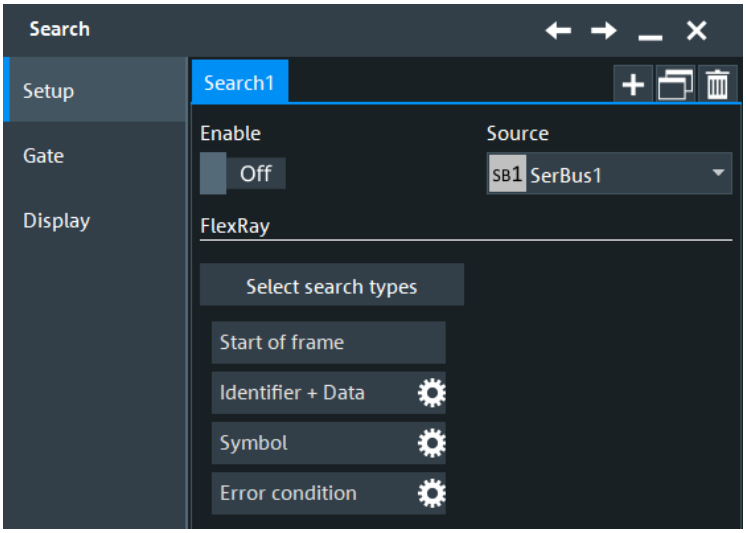
Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 11, "Search functions", on page 419.

13.9.4.1 FlexRay search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"



Select Search Types

Selects one criterion or an AND-combination of criteria to be searched for. If more than one criterion is selected, all criteria must be fulfilled by a frame for it to be shown in the search results.

"Start of frame" Searches for the first rising edge after the transmission start sequence (TSS).

"Identifier+data"

Searches for the decoded frame content, on header and payload data:

- Indicator bits, see "Indicator bits" on page 631
- Frame identifier, see "Frame ID (min/max)" on page 632
- Payload length, see "Payload length (min/max)" on page 632
- Cycle count, see "Cycle count (min, max), Step" on page 632
- Data position, see "Position, Index (min, max) - Data setup" on page 633
- Data bit pattern, see "Condition, Data (min, max) - Data setup" on page 633

"Symbol" Searches for a symbol or wakeup pattern, see "Symbol" on page 633.

"Error condition"

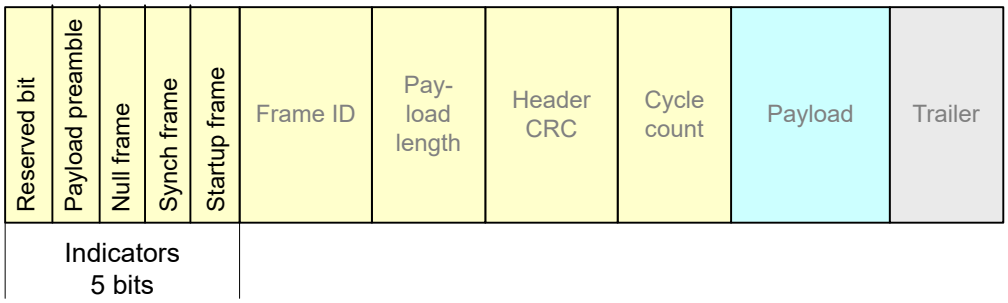
Searches for one or more errors that are detected in the decoded data, see "Error Condition" on page 633.

Remote command:

`SEARch:TRIGger:FLXRay[:SSOFrame]` on page 1885
`SEARch:TRIGger:FLXRay:SSYMbol` on page 1885
`SEARch:TRIGger:FLXRay:SIDData` on page 1885
`SEARch:TRIGger:FLXRay:SERRor` on page 1884

Indicator bits

Searches for one or more indicator bits at the beginning of the header segment. Each bit can be set to 0, 1, or X (do not care).



"Payload preamble"	Indicates a Network Management Vector in the payload segment. The NMV allows the host processor to send data directly, without processing by the communication controller.
"Null frame"	Indicates a frame without usable data.
"Sync frame"	Indicates that the frame is used for synchronization of the FlexRay system. Only sync nodes can send this frame type.
"Startup frame"	Indicates a startup frame used for startup of the network. Only specific start nodes can send this frame type.

Remote command:
`SEARch:TRIGger:FLXRay:PLPReamble` on page 1890
`SEARch:TRIGger:FLXRay:NUFRame` on page 1890
`SEARch:TRIGger:FLXRay:SYFRame` on page 1892
`SEARch:TRIGger:FLXRay:STFRame` on page 1891

Frame ID (min/max)
The frame ID contains the number of the slot in which the frame is transmitted. Each frame ID occurs only once during a FlexRay cycle.

The setup conditions are the same as in the FlexRay trigger setup, see "Frame ID (min/max)" on page 626.

Remote command:
`SEARch:TRIGger:FLXRay:FCONdition` on page 1889
`SEARch:TRIGger:FLXRay:FMIN` on page 1889
`SEARch:TRIGger:FLXRay:FMAX` on page 1889

Payload length (min/max)
The payload length contains the number of words transmitted in the payload segment. Information is transmitted in 2-byte words, so the number of data bytes in the payload segment is twice the payload length.

The setup conditions are the same as in the FlexRay trigger setup, see "Payload length (min/max)" on page 626.

Remote command:
`SEARch:TRIGger:FLXRay:PCONdition` on page 1890
`SEARch:TRIGger:FLXRay:PMIN` on page 1891
`SEARch:TRIGger:FLXRay:PMAX` on page 1891

Cycle count (min, max), Step
The cycle count contains the number of the current FlexRay cycle.

The setup conditions are the same as in the FlexRay trigger setup, see "Cycle count (min, max), Step" on page 626.

Remote command:
`SEARch:TRIGger:FLXRay:CENable` on page 1885
`SEARch:TRIGger:FLXRay:CMIN` on page 1886
`SEARch:TRIGger:FLXRay:CMAX` on page 1886
`SEARch:TRIGger:FLXRay:CSTep` on page 1886

Position, Index (min, max) - Data setup
Sets the position of the first byte of data bit pattern within the payload segment. You can define an exact position, or a position range.

The setup conditions are the same as in the FlexRay trigger setup, see "Position, Index (min, max) - Data setup" on page 627.

Remote command:
`SEARch:TRIGger:FLXRay:DPOPerator` on page 1888
`SEARch:TRIGger:FLXRay:DPOSition` on page 1888
`SEARch:TRIGger:FLXRay:DPTO` on page 1888

Condition, Data (min, max) - Data setup
Specifies the data bit pattern to be found in the payload segment. The starting point of the pattern is defined by "Position, Index (min, max) - Data setup" on page 633. The pattern comparison is byte-aligned, and the instrument triggers at the end of a byte.

The setup conditions are the same as in the FlexRay trigger setup, see "Condition, Data (min, max) - Data setup" on page 627.

Remote command:
`SEARch:TRIGger:FLXRay:DCONdition` on page 1887
`SEARch:TRIGger:FLXRay:DMIN` on page 1887
`SEARch:TRIGger:FLXRay:DMAX` on page 1887

Symbol
Searches for a symbol or a wakeup pattern.

"CAS/MTS"	Collision Avoidance Symbol / Media access Test Symbol. These symbols are identical and can be sent in the optional symbol window at the end of a communication cycle. They are used to avoid collisions during the system start.
"Wakeup Pattern"	The wakeup pattern is sent to activate the nodes of the system.

Remote command:
`SEARch:TRIGger:FLXRay:SYMBol` on page 1892

Error Condition
Selects the error type to be searched for. You can select one or more error types as search condition.

The error types are the same as in the FlexRay trigger setup, see "Error conditions" on page 628.

Remote command:
`SEARch:TRIGger:FLXRay:BSSerror` on page 1892
`SEARch:TRIGger:FLXRay:FESerror` on page 1892
`SEARch:TRIGger:FLXRay:FSSerror` on page 1893
`SEARch:TRIGger:FLXRay:HCRCerror` on page 1893
`SEARch:TRIGger:FLXRay:PCRCerror` on page 1893

13.9.4.2 FlexRay search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands:

- `SEARch:RESult:FLXRay:FCOunt?` on page 1894
- `SEARch:RESult:FLXRay:FRAMe<m>:ADID?` on page 1894
- `SEARch:RESult:FLXRay:FRAMe<m>:CSSTate?` on page 1894
- `SEARch:RESult:FLXRay:FRAMe<m>:CSValue?` on page 1895
- `SEARch:RESult:FLXRay:FRAMe<m>:CYCount?` on page 1895
- `SEARch:RESult:FLXRay:FRAMe<m>:DATA?` on page 1895
- `SEARch:RESult:FLXRay:FRAMe<m>:FCSTate?` on page 1896
- `SEARch:RESult:FLXRay:FRAMe<m>:FCValue?` on page 1896
- `SEARch:RESult:FLXRay:FRAMe<m>:FLAGs?` on page 1896
- `SEARch:RESult:FLXRay:FRAMe<m>:PAYLength?` on page 1897
- `SEARch:RESult:FLXRay:FRAMe<m>:STATus?` on page 1897
- `SEARch:RESult:FLXRay:FRAMe<m>:STARt?` on page 1898
- `SEARch:RESult:FLXRay:FRAMe<m>:STOP?` on page 1898
- `SEARch:RESult:FLXRay:FRAMe<m>:SYMBol?` on page 1898
- `SEARch:RESult:FLXRay:FRAMe<m>:TYPE?` on page 1898

13.10 SENT (option R&S RTO6-K520)

Single Ended Nibble Transmission (SENT) is a serial transmission interface protocol originally specified for the communication of sensors and control units in automotive electronics.

SENT is a protocol standard governed by Society of Automotive Engineers (SAE J2716). For detailed information, refer to the SENT standard specification on http://www.sae.org.

The SENT protocol is used exclusively in automotive applications, as for example electrical power steering, advanced driver assistance like parking assist or sensing of pressure, throttle position, pedal position, airflow mass, liquid level, etc.

The R&S RTO6 option R&S RTO6-K520 provides serial triggering, decoding and a highly customizable search on decoded SENT signals.

13.10.1 The SENT protocol

This chapter provides an overview of the protocol characteristics, encoding scheme, identifiers and trigger possibilities.

The SENT protocol transmits signal values point-to-point from a sensor to a controller (electronic control unit ECU), unidirectional. In contrast to conventional measurements, you can receive multiple data parameters via the SENT interface in a single transmission. Nevertheless, SENT is characterized by its simplicity and yet very high customizability to meet the individual requirements of the applications.

SENT operates via a three wire connection, a signal line, a supply voltage line for the sensor and a ground line. It transmits data digitally in variable timing units and evaluates the time between two falling edges (single edges). The signal is amplitude modulated with a constant amplitude voltage. Thus influences of interfering signals are not critical.

SENT key features

Main characteristics of SENT are:

- serial communication protocol
- 3 wires: SENT (signal line), 5V (voltage line), GND (ground line)
- output only, from sensor to receiver
- point-to-point transmission, no bus
- digital transmission
- high baud rate
- data transmission in variable timing units of 4 bits (1 nibble) between two falling edges
- transmitter-specific clock period (tick)
- time measured between single falling edges

13.10.1.1 SENT transmission concept

A sensor converts the analog measured data to a digital signal, and thus transmits a series of pulses to the receiver. The receiver, e.g. an ECU processes the received signal also digitally.

The format of a SENT message frame has a fixed pulse order and a transmitter-specific clock period. The total transmission time varies depending on the clock variation of the transmitter and the transmitted data values. The data pulses embedded in the transmission sequence represent one or multiple data parameters to be communicated. The last pulses in a message frame are the CRC check pulse, allowing the receiver to perform a number of diagnostic tests, and an optional pause pulse.

A SENT transmission starts without a request from the receiver. Consecutive sequences are transmitted continuously after the falling edge of the last pulse.

The SENT protocol distinguishes between two channel types:

- **Fast channel:** transmits primary data, i.e. sensor readings like temperature, pressure, mass air flow, throttle position.
- **Slow channel:** transmits secondary data consisting of transfer characteristics, sensor ID, type, manufacturer diagnostic, etc.
  The slow channel transmission provides two serial message formats *Short* and *Enhanced* for customizing the secondary data.

The data of both, the fast and the slow channels is transmitted simultaneously, by including two bits of a slow channel message in the message frame of the fast channel. Even though it requires many fast channel messages to complete a slow channel message, you can use this function to transmit several slow channel messages with minimal impact on the primary sensor data and the data rate.

13.10.1.2 SENT message definitions

SENT terms

See the specific terms and definition used in SENT protocol:

- **Tick (clock tick):** basic unit of time
  - transmitter-specific nominal clock period
  - 3 µs < clock tick <90 µs, with max. 20 % clock variation
- **Nibble:** minimum unit of data
  - used to transmit data
  - variable timing units between two falling edges

SENT fast channel

The SENT protocol enables you to transmit measurements of multiple sensors in one transmission sequence with data signals of varying length. The diagram in Figure 13-17 shows, for example, the encoding scheme for two 12-bit data signals.

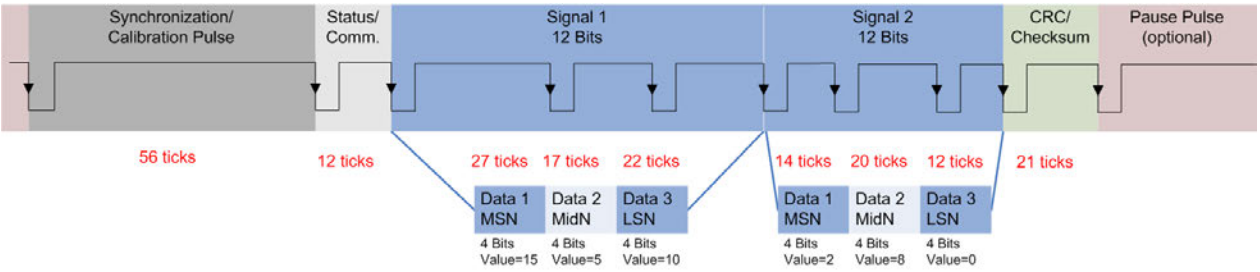


Figure 13-17: Example of a SENT transmission sequence

The format of a SENT transmission sequence consists of the following pulses:

- **Synchronization/Calibration Pulse:**
  - initial sequence of the receiver
  - the start condition is the falling edge of the last pulse (CRC or Pause)
  - nominal pulse period is 56 clock ticks
  - measures the actual clock variation of the transmitter and calculates the tick timing
- **Status/Communication Pulse (Nibble)**
  - one 4 bit pulse
  - communicates status and enables the sensor to include slow channel message bits
    0: (LSB) specific application
    1: specific application
    2: Serial Data message or specific application (e.g. Infineon TLE4998S)
    3: (MSB) 1= message start; 0=Serial Data message or specific application (e.g. Infineon TLE4998S)
  - 12 to 27 clock ticks
  - not included in CRC frame calculation
- **Data Pulses (Nibbles)**
  - one up to six 4 bit data nibbles
  - 12 to 27 clock ticks pulse period
  - initial logic 0 time with ≥5 ticks, subsequent logical 1 with variable duration
- **CRC/Checksum**
  - one 4 bit pulse
  - used for error checking of data nibbles (status nibble not included)
  - detects single bit, odd number of nonconsecutive and single burst errors
- **Pause Pulse**
  - one optional pulse
  - variable pulse length: 12 to 768 clock ticks
  - can be used to create a transmission with constant number of clock ticks

SENT slow channel

Short Serial Messages

For transmission of a slow channel message, 2 bits are included in a fast channel message, see the status nibble (Bit 2,3) in Figure 13-18.

A short serial message needs 16 fast channel messages until it is completely transmitted. Prerequisite for the complete transmission of the slow channel message are 16 consecutive error-free fast channel transmissions.

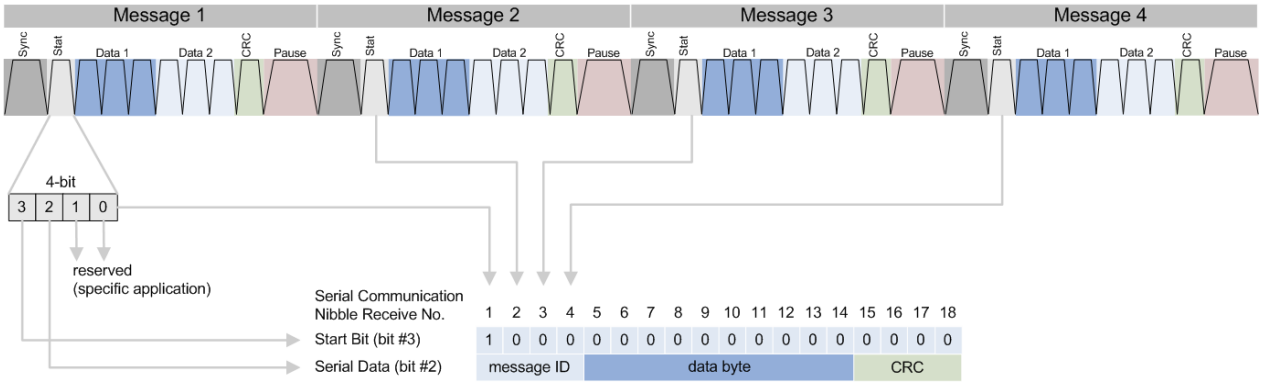


Figure 13-18: One serial message, composed of 16 SENT consecutive fast channel transmissions

Enhanced Serial Messages

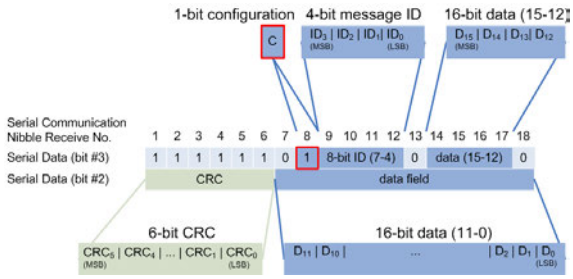
The transmission of an enhanced serial message format requires 18 fast channel transmissions. Each slow channel message is assigned a message ID, which is transmitted with the data.

The enhanced serial message format provides two alternatives for configuring the message:

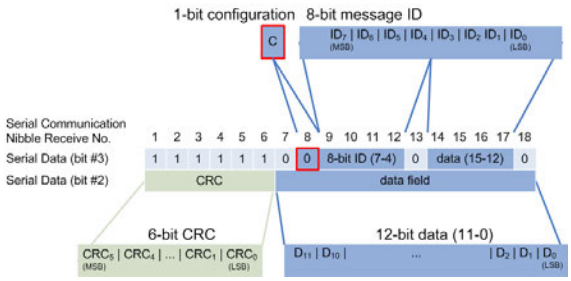
- 4 bit ID and 16 bit data
- 8 bit ID and 12 bit data

The graphs below illustrate the variants.

Table 13-14: Enhanced serial message formats



16 bit data and 4 bit message ID



12 bit data and 8 bit message ID

Trigger

The R&S RTO6 can trigger on various parts of SENT pulses. The data line must be connected to an input channel, triggering on math and reference waveforms is not possible.

SENT enables you to trigger on:

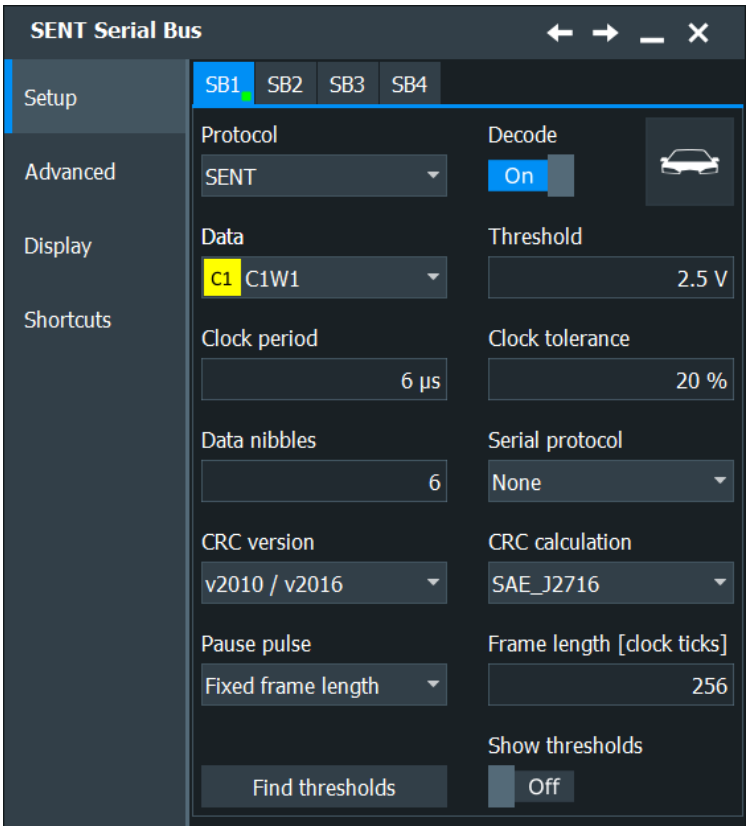
- Calibration/synchronization pulse
- Transmission sequence
- Serial messages
- Error conditions

13.10.2 SENT configuration

13.10.2.1 SENT configuration

Access: [Bus] > "Setup" tab > "Protocol" = *SENT*

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Data
Sets the source of the data line.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Remote command:
`BUS<m>:SENT:DATA:SOURce` on page 1899

Threshold
Sets the threshold value for digitization of the signal. If the signal value on the line is higher than the threshold, the signal state is high. Otherwise, the signal state is considered low if the signal value is below the threshold.

There are several ways to set the threshold:
- "Threshold"
  Enter the value directly in the field.
- "Autoset"
  Starts software algorithms for determining the signal threshold levels and bitrate.
- "Find thresholds"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
`BUS<m>:SENT:DATA:THReshold` on page 1899
`BUS<m>:SENT:TECHnology` on page 1900
`BUS<m>:SETReflevels` on page 1671
`BUS<m>:FAUToset` on page 1671

Clock Period
Sets the transmitter-specific nominal clock period (clock tick).

The clock period and signal length determine the speed of transmission.

Remote command:
`BUS<m>:SENT:CLKPeriod` on page 1900

Clock tolerance
Specifies a tolerated deviation of the clock.

Remote command:
`BUS<m>:SENT:CLKTolerance` on page 1900

Data Nibbles
Sets the number of data units in a single transmission sequence.

The maximum number of data nibbles is 6.

Remote command:
`BUS<m>:SENT:DNIBbles` on page 1901

Serial protocol
Selects the protocol format in the transmitted signal.

"Short" Short serial messages.

"Enhanced" Enhanced serial messages.

"None" No serial messages. Transmission sequences only.

Remote command:
`BUS<m>:SENT:SFORmat` on page 1901

CRC version
Selects the version the CRC check is based on.

"Legacy" Based on the crc calculation version used earlier than 2010.

"v2010" Based on the recent crc calculation version updated in 2010.

Remote command:
`BUS<m>:SENT:CRCVersion` on page 1901

CRC calculation
Selects the method for CRC calculation.

"SAE_J2716" Calculates the CRC according to the SAE standard.
For this method the checksum is calculated over all nibbles except the communication and status nibble.

"TLE_4998X" Calculates the CRC according to the standard computing method for Infineon TLE_4998X sensors.
For this method the checksum is calculated over all nibbles except the communication nibble.

Remote command:
`BUS<m>:SENT:CRCMethod` on page 1901

Pause pulse
Determines whether a pause pulse is transmitted after the checksum nibble.

You can use this pulse to create a transmission with a constant number of clock ticks. The pause pulse length can be between a minimum of 12 clock ticks up to 768 (3*256) ticks at a maximum.

"No"
No pause pulse between the transmission sequences.

"Yes" Pause pulse with fixed length at the end of each transmission sequence.
The R&S RTO6 computes the length of the pause pulse automatically.

"For constant frame length"
Pause pulse with dynamic length to maintain a fixed transmission sequence length.
To define the constant frame length, set the number of clock ticks under Frame length [clock ticks].

Remote command:
`BUS<m>:SENT:PPULse` on page 1902

Frame length [clock ticks]
Determines the frame length in terms of ticks. The dialog displays this settings parameter, if the signal has a constant frame length.

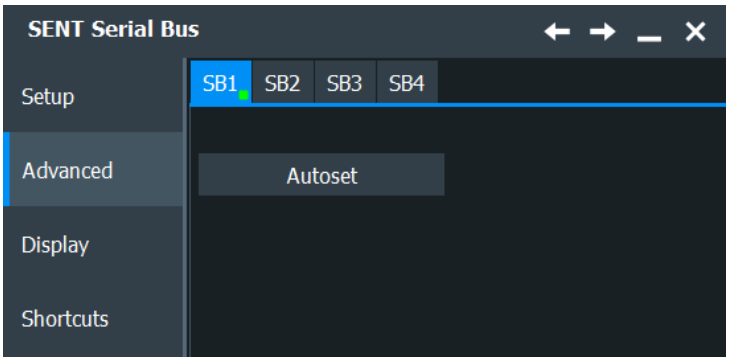
Remote command:
`BUS<m>:SENT:PPFLength` on page 1902

Show thresholds
If enabled, the threshold levels are displayed in the diagram.

Remote command:
`BUS<m>:THReshold` on page 1671

13.10.2.2 Advanced settings

Access: [Bus] > "Advanced" tab.



Autoset
Starts software algorithms for determining the signal threshold levels and bitrate.

Remote command:
`BUS<m>:FAUToset` on page 1671

13.10.2.3 Configuring SENT protocol

The configuration of the SENT is simple - assign the two lines to input channels, and set the thresholds.

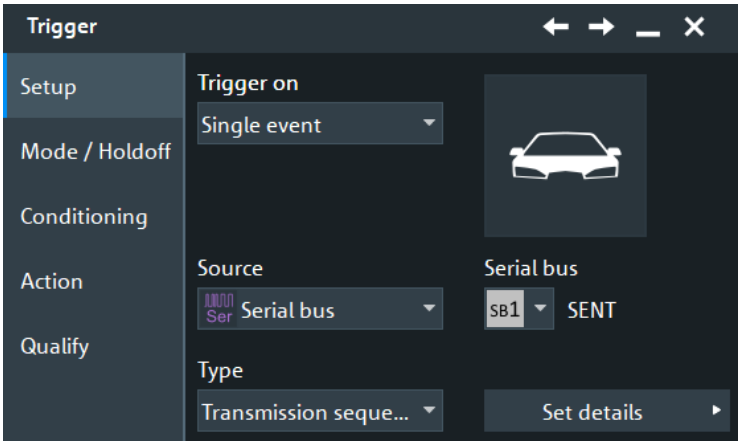
For details on configuration settings, see Chapter 13.10.2.1, "SENT configuration", on page 639.

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. Tap the "Protocol" button and select the protocol: "SPI".
4. Tap the "Data" button, and select the waveform of the data line.
5. If necessary, set the "Transmission Setup" parameters.
6. Set the logical thresholds.
7. Enable "Decode".

13.10.3 SENT trigger

13.10.3.1 SENT trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Type

Selects the event of the SENT transmission sequence or message to be triggered on.

The RTO6 triggers always on the falling edge of a pulse, i.e. at the end of the selected type nibble in a transmission sequence.

"Calibration or Sync"

Triggers at the end of the "Synchronization/Calibration" pulse.

This setting does not require any input parameters.

"Transmission Sequence"

Triggering depends on the additional selectable "Sequence" parameter:

- "Sequence > Status": triggers at the end of the "Status" nibble.

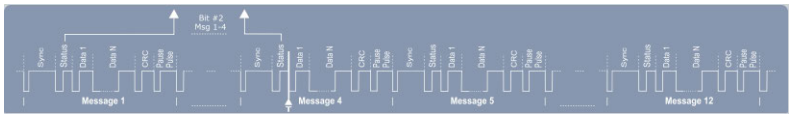
- "Sequence > Status+Data": triggers at the end of the last data nibble.

Description of trigger type specific settings: "Transmission Sequence setup" on page 645

"Serial Message"

Triggering on a serial message depends on the serial protocol selected with Serial protocol > Short | Enhanced and the associated setting parameters:

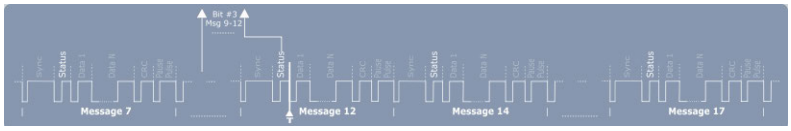
- "Short" serial message
  - "Sequence > Identifier": triggers at the end of the "ID".



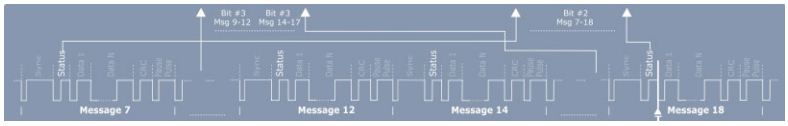
  - "Sequence > ID+Data": triggers at the end of the "ID and Data".

  Description of trigger type specific settings: "Serial Message setup" on page 646

- "Enhanced" serial message
  - "Sequence > Identifier": triggers at the end of the "ID".



  - "Sequence > ID+Data": triggers at the end of the "ID and Data".



  Description of trigger type specific settings: "Serial Message setup" on page 646

"Error condition"

Triggers if certain errors occur.

You can select the following error events for triggering:

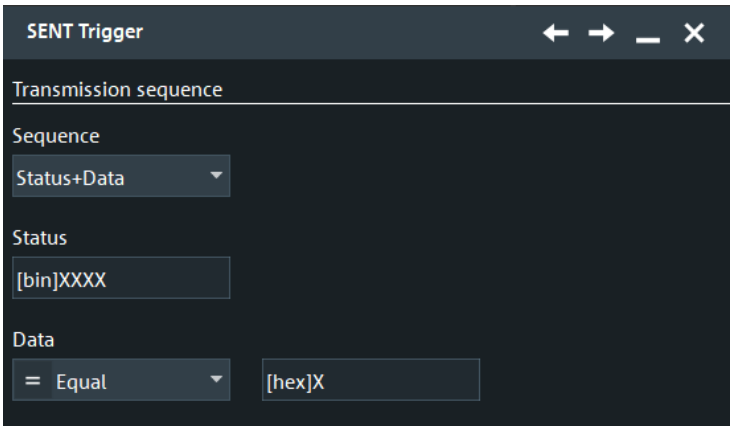
- "Form Error"
- "Calibration Pulse Error"
- "Pulse Period Error"
- "CRC Error"
- "Irregular Frame Length Error"

Description of error-specific trigger conditions, see "Error conditions setup" on page 648.

Remote command:
`TRIGger<m>:SENT:TYPE` on page 1903

Transmission Sequence setup
Configures the trigger conditions for trigger type transmission sequence.

Note: The displayed parameters depend on the selected "Sequence". The instrument displays the data setting parameters when you select "Status+Data", see "Type" on page 643.



Sequence ← Transmission Sequence setup
Selects the condition for triggering in a single transmission sequence.

You can trigger on the end of the status nibble or the combination of the status and data nibbles.

Remote command:
`TRIGger<m>:SENT:TTYPe` on page 1904

Status ← Transmission Sequence setup
Defines the data bits for the status nibble.

Remote command:
`TRIGger<m>:SENT:STATus` on page 1904

Condition ← Transmission Sequence setup
Selects the operator to define a specific data pattern or a data range.

The available operators:
- Equal, Not equal
- Less than, Greater than
- Less or equal, Greater or equal
- In range, Out of range

Remote command:
`TRIGger<m>:SENT:TDCN` on page 1905

Nibbles ← Transmission Sequence setup
Displays the number of data nibbles of the transmission sequence.

Remote command:
`BUS<m>:SENT:DNIBbles` on page 1901

Data min ← Transmission Sequence setup
Sets the data pattern. Enter the bytes in msb first bit order. The maximum pattern length is 64 bit. Waveform data is compared with the pattern byte-by-byte.

Remote command:
`TRIGger<m>:SENT:TDMN` on page 1905

Data max ← Transmission Sequence setup
The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:SENT:TDMX` on page 1905

Serial Message setup
Configures the trigger conditions for a serial message.

Note: The displayed parameters depend on the selected "Sequence". The instrument indicates the data setting parameters when you select "ID+Data", see "Type" on page 643.

Sequence ← Serial Message setup
Selects the condition for triggering in a serial message.

You can trigger on the end of an identifier nibble or the combination of the identifier and data nibble(s).

Remote command:
`TRIGger<m>:SENT:STYPe` on page 1905

ID Type ← Serial Message setup
Selects the message ID format for the enhanced serial message type.

You can select either 4-bit or 8-bit message ID.

Remote command:
`TRIGger<m>:SENT:SIDType` on page 1906

Identifier Condition ← Serial Message setup
The available operators:
- Equal, Not equal

- Less than, Greater than
- Less or equal, Greater or equal
- In range, Out of range

Remote command:
`TRIGger<m>:SENT:SICN` on page 1906

Identifier min ← Serial Message setup
Defines the bit pattern of the message identifier.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

Remote command:
`TRIGger<m>:SENT:SIMN` on page 1906

Identifier max ← Serial Message setup
The second identifier pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:SENT:SIMX` on page 1906

Data Condition ← Serial Message setup
Selects the operator to set a specific data pattern or a data range.

The available operators:
- Equal, Not equal
- Less than, Greater than
- Less or equal, Greater or equal
- In range, Out of range

Remote command:
`TRIGger<m>:SENT:SDCN` on page 1907

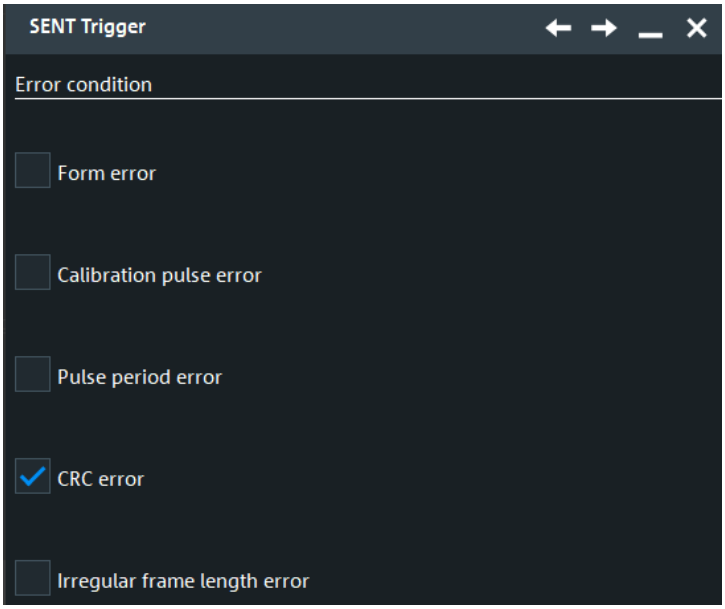
Data min ← Serial Message setup
Sets the data pattern. Enter the bytes in msb first bit order. The maximum pattern length is 64 bit. Waveform data is compared with the pattern byte-by-byte.

Remote command:
`TRIGger<m>:SENT:SDMN` on page 1907

Data max ← Serial Message setup
The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:SENT:SDMX` on page 1907

Error conditions setup



Executes a trigger event if one or more of the following errors occur:

- "Form Error"
  Detects a form error in serial messages (short and enhanced).
  An error occurs when at least one of the transmission sequences that form a serial message has an error.
- "Calibration Pulse Error"
  Detects a calibration pulse error in transmission sequences.
  An error occurs when:
  – the duration of the "Calibration/Sync" pulse (in ticks) is less than 56*(1-clock tolerance) or more than 56*(1+clock tolerance)
  – the "Calibration/Sync" pulse duration of frame (–1) varies by more than 1.5625% from the "Calibration/Sync" pulse duration of frame (n)
- "Pulse Period Error"
  Detects an error in the "Calibration/Sync" pulse in transmission sequences.
  An error occurs when a nibble has any of the following:
  – Number of ticks at low is fewer than 4 ticks.
  – Nibble value < 0 (fewer than 12 ticks) or > 15 (more than 27 ticks).
- "CRC Error"
  Detects a checksum error in both, the transmission sequences and serial messages.
  The CRC length is 4 bits for transmission sequences and short serial messages, and 6 bit of enhanced serial messages.
- "Irregular Frame Length Error"
  Detects frame length errors in transmission sequences when pause pulse mode constant frame length is set, see "Pause pulse" on page 641.
  A frame length error occurs, when the total length of the transmission sequence (including pause pulse) does not match the frame length setting, see "Frame length [clock ticks]" on page 641.

Remote command:
`TRIGger<m>:SENT:FORMerror` on page 1907
`TRIGger<m>:SENT:PULSeerror` on page 1908
`TRIGger<m>:SENT:PPERioderror` on page 1908
`TRIGger<m>:SENT:CRCerror` on page 1908
`TRIGger<m>:SENT:IRFLength` on page 1909

13.10.3.2 Triggering on SENT signals

Prerequisites: A SENT bus is configured, see Chapter 13.10.2.3, "Configuring SENT protocol", on page 642.

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to SENT.
5. Select the "Trigger type".
6. For more complex trigger types, set the parameters according to the selected "Trigger type".
   For details, see Chapter 13.10.3.1, "SENT trigger settings", on page 643.

13.10.4 SENT label list

SENT label lists provide a very useful way of translating the decoded data into user format. The label lists are highly customizable. The format of supplying the label list description is through a `.xml` file and is explained with an example, see "Label List Structure for SENT Protocol" on page 649.

See also: Chapter 13.1.4, "Content and format of the PTT file", on page 489.

Label List Structure for SENT Protocol

```
<sb:FRAME NAME="Diagnostic Error Codes" STATE="ON">
  <!-- Start of a Frame Definition -->
  <!-- This block defines the information of a Transmission Sequence
   or Serial Message:
   NAME => Symbolic Label of the Frame
   STATE [ON/OFF] => When ON, this frame Translation is taken into consideration.
                     When OFF, this frame Translation is skipped.-->
<sb:DESCRIPTION> used to diagnose the current SENT System</sb:DESCRIPTION>
  <!-- Doesn't affect the Translation -->
<sb:ID-VALUE>01</sb:ID-VALUE>
  <!-- ID Value of the Serial Message (in decimal) -->
  <!-- Absence of the ID-VALUE field implies that the current Frame Translation
   is to be used for Transmission Sequences and not for a Serial Message -->
<sb:ID-LENGTH>8</sb:ID-LENGTH>
```

```
  <!-- ID Length of the Serial Message (in bits) -->
<sb:DATA-SIZE>12</sb:DATA-SIZE>
  <!-- Data Length of the Serial Message (in bits) -->
<sb:SIGNALS>
  <!-- This block defines the information of the Signals embedded
   in the Data Field of the Frame (Transmission Sequence or Serial Message) -->
<sb:SIGNAL ID="Diagnostic">
  <!-- Unique ID of the Signal (no effect on Translation) -->
<sb:SHORT-NAME>Diagnostic Code</sb:SHORT-NAME>
  <!-- Name of the Signal -->
<sb:DESCRIPTION></sb:DESCRIPTION>
  <!-- Info Field (no effect on Translation) -->
<sb:BIT-POSITION>11</sb:BIT-POSITION>
  <!-- Starting Bit position of the Signal
   (The whole Data Field is represented as MSB -> LSB Sequence) -->
<sb:BIT-LENGTH>12</sb:BIT-LENGTH>
  <!-- Number of Bits representing the Signal Value -->
<sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
  <!-- Byte Order of the Signal Value [MSB or LSB], Default: MSB -->
<sb:VALUE-TYPE>ENUM</sb:VALUE-TYPE>
  <!-- Representation of the Bits [ENUM, UNSIGNED_INT, INT, FLOAT, DOUBLE],
   Default: UNSIGNED_INT
   The Signal Value is calculated according to the following:
   Translated_Value = Encoded_Value * FACTOR + OFFSET -->
<sb:FACTOR>1.0</sb:FACTOR>
  <!-- Signal Factor (decimal value)-->
<sb:OFFSET>0.0</sb:OFFSET>
  <!-- Signal Offset (decimal value)-->
<sb:MIN>0</sb:MIN>
  <!-- Minimum Signal Value (decimal value) -->
<sb:MAX>4096</sb:MAX>
  <!-- Maximum Signal Value (decimal value) -->
<sb:ENUM-VALUES>
  <!-- This block is only valid (and taken into consideration)
   when the VALUE-TYPE is ENUM
   It defines the Enumeration List Translation of the Signal -->
<sb:ENUM INDEX="0" LABEL="No Error"/>
   <!-- INDEX is the Enum Value (corresponds to the Signal Value in decimal),
    LABEL is the matching Translated Signal Value -->
<sb:ENUM INDEX="1" LABEL="Channel 1 out of range high"/>
</sb:ENUM-VALUES>
  <!-- End of Signal Enumeration List Definition -->
</sb:SIGNAL>
  <!-- End of a Signal Definition -->
  <!-- More Signals can be defined here! -->
</sb:SIGNALS>
  <!-- End of list of Signals Definition -->
</sb:FRAME>
  <!-- End of Frame Definition -->
```

For an example to label list translation, see Chapter 13.10.4.1, "SENT label list translation example", on page 651.

Remote command:

- `BUS<m>:SENT:FRAMe<n>:SYMBol?` on page 1914
- `BUS<m>:SENT:FRAMe<n>:SDATa?` on page 1914
- `BUS<m>:SENT:FRAMe<n>:SDEXport?` on page 1914

13.10.4.1 SENT label list translation example

The example shows the `xml` sequence for a label list translation in the SENT protocol:

```
<?xml version="1.0" encoding="UTF-8"?>
<sb:LABEL-LIST-FILE>
 <sb:PROJECT ID="SENT-TRANSLATION SYSTEM">
  <sb:SHORT-NAME>SENT</sb:SHORT-NAME>
  <sb:LONG-NAME>SENT-Translation System Demo</sb:LONG-NAME>
  <sb:DESCRIPTION>This is the database for Translation demo for SENT.</sb:DESCRIPTION>
 </sb:PROJECT>
 <sb:FRAMES>
  <sb:FRAME NAME="Air Temperature" STATE="ON">
   <sb:DESCRIPTION></sb:DESCRIPTION>
   <sb:ID-VALUE>0</sb:ID-VALUE>
   <sb:ID-LENGTH>4</sb:ID-LENGTH>
   <sb:DATA-SIZE>16</sb:DATA-SIZE>
  </sb:FRAME>
  <sb:FRAME NAME="Humidity" STATE="OFF">
   <sb:DESCRIPTION></sb:DESCRIPTION>
   <sb:ID-VALUE>2</sb:ID-VALUE>
   <sb:ID-LENGTH>4</sb:ID-LENGTH>
   <sb:DATA-SIZE>16</sb:DATA-SIZE>
  </sb:FRAME>
  <sb:FRAME NAME="Barometric Pressure" STATE="OFF">
   <sb:DESCRIPTION></sb:DESCRIPTION>
   <sb:ID-VALUE>4</sb:ID-VALUE>
   <sb:ID-LENGTH>4</sb:ID-LENGTH>
   <sb:DATA-SIZE>16</sb:DATA-SIZE>
  </sb:FRAME>
  <sb:FRAME NAME="Configuration Code" STATE="OFF">
   <sb:DESCRIPTION></sb:DESCRIPTION>
   <sb:ID-VALUE>04</sb:ID-VALUE>
   <sb:ID-LENGTH>8</sb:ID-LENGTH>
   <sb:DATA-SIZE>12</sb:DATA-SIZE>
  </sb:FRAME>
  <sb:FRAME NAME="Manufacturer Code" STATE="OFF">
   <sb:DESCRIPTION></sb:DESCRIPTION>
   <sb:ID-VALUE>05</sb:ID-VALUE>
   <sb:ID-LENGTH>8</sb:ID-LENGTH>
   <sb:DATA-SIZE>12</sb:DATA-SIZE>
```

```
</sb:FRAME>
<sb:FRAME NAME="Sensor Type" STATE="ON">
 <sb:DESCRIPTION>specifies the SENT Sensor Type</sb:DESCRIPTION>
 <sb:ID-VALUE>03</sb:ID-VALUE>
 <sb:ID-LENGTH>8</sb:ID-LENGTH>
 <sb:DATA-SIZE>12</sb:DATA-SIZE>
 <sb:SIGNALS>
  <sb:SIGNAL ID="Sensor Class">
   <sb:SHORT-NAME>Sensor Class</sb:SHORT-NAME>
   <sb:BIT-POSITION>11</sb:BIT-POSITION>
   <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
   <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
   <sb:VALUE-TYPE>ENUM</sb:VALUE-TYPE>
   <sb:FACTOR>1.0</sb:FACTOR>
   <sb:OFFSET>0.0</sb:OFFSET>
   <sb:MIN>0</sb:MIN>
   <sb:MAX>32.0</sb:MAX>
   <sb:ENUM-VALUES>
    <sb:ENUM INDEX="0" LABEL="Not Specified"/>
    <sb:ENUM INDEX="1" LABEL="P"/>
    <sb:ENUM INDEX="2" LABEL="P/-"/>
    <sb:ENUM INDEX="3" LABEL="P/S"/>
    <sb:ENUM INDEX="4" LABEL="P/S/Default T"/>
    <sb:ENUM INDEX="5" LABEL="P/S/Sensor-Specific T"/>
    <sb:ENUM INDEX="6" LABEL="P1/P2"/>
    <sb:ENUM INDEX="7" LABEL="P/Default T"/>
    <sb:ENUM INDEX="8" LABEL="P/Sensor-Specific T"/>
    <sb:ENUM INDEX="9" LABEL="P1/P2/Default T"/>
    <sb:ENUM INDEX="10" LABEL="P1/P2/Sensor-Specific T"/>
    <sb:ENUM INDEX="16" LABEL="Not Defined"/>
    <sb:ENUM INDEX="17" LABEL="MAF (hi-res,lin)"/>
    <sb:ENUM INDEX="18" LABEL="MAF (hi-res,non-lin)"/>
    <sb:ENUM INDEX="19" LABEL="MAF (hi-res,lin) / Pressure"/>
    <sb:ENUM INDEX="20" LABEL="MAF (hi-res,non-lin) / Pressure"/>
    <sb:ENUM INDEX="21" LABEL="MAF (lin) / Pressure (hi-res)"/>
    <sb:ENUM INDEX="22" LABEL="MAF (non-lin) / Pressure (hi-res)"/>
   </sb:ENUM-VALUES>
  </sb:SIGNAL>
 </sb:SIGNALS>
</sb:FRAME>
<sb:FRAME NAME="SENT Standard Revision" STATE="ON">
 <sb:SHORT-NAME>SENT Standard</sb:SHORT-NAME>
 <sb:DESCRIPTION>specifies the SENT Standard Revision Number</sb:DESCRIPTION>
 <sb:ID-VALUE>10</sb:ID-VALUE>
 <sb:ID-LENGTH>4</sb:ID-LENGTH>
 <sb:DATA-SIZE>8</sb:DATA-SIZE>
 <sb:SIGNALS>
  <sb:SIGNAL ID="Revision">
   <sb:SHORT-NAME>Revision</sb:SHORT-NAME>
   <sb:DESCRIPTION>SENT-Standard Revision Number</sb:DESCRIPTION>
```

```
    <sb:BIT-POSITION>7</sb:BIT-POSITION>
    <sb:BIT-LENGTH>8</sb:BIT-LENGTH>
    <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
    <sb:VALUE-TYPE>ENUM</sb:VALUE-TYPE>
    <sb:FACTOR>1.0</sb:FACTOR>
    <sb:OFFSET>0.0</sb:OFFSET>
    <sb:MIN>0</sb:MIN>
    <sb:MAX>4.0</sb:MAX>
    <sb:ENUM-VALUES>
     <sb:ENUM INDEX="0" LABEL="Not defined"/>
     <sb:ENUM INDEX="1" LABEL="J2716 Rev 1"/>
     <sb:ENUM INDEX="2" LABEL="J2716 Rev 2"/>
     <sb:ENUM INDEX="3" LABEL="J2716 Rev 3"/>
    </sb:ENUM-VALUES>
   </sb:SIGNAL>
  </sb:SIGNALS>
 </sb:FRAME>
 <sb:FRAME NAME="Diagnostic Error Codes" STATE="ON">
  <sb:DESCRIPTION>used to diagnose the current SENT System</sb:DESCRIPTION>
  <sb:ID-VALUE>01</sb:ID-VALUE>
  <sb:ID-LENGTH>8</sb:ID-LENGTH>
  <sb:DATA-SIZE>12</sb:DATA-SIZE>
  <sb:SIGNALS>
   <sb:SIGNAL ID="Diagnostic">
    <sb:SHORT-NAME>Diagnostic Code</sb:SHORT-NAME>
    <sb:DESCRIPTION></sb:DESCRIPTION>
    <sb:BIT-POSITION>11</sb:BIT-POSITION>
    <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
    <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
    <sb:VALUE-TYPE>ENUM</sb:VALUE-TYPE>
    <sb:FACTOR>1.0</sb:FACTOR>
    <sb:OFFSET>0.0</sb:OFFSET>
    <sb:MIN>0</sb:MIN>
    <sb:MAX>4096</sb:MAX>
    <sb:ENUM-VALUES>
     <sb:ENUM INDEX="0" LABEL="No Error"/>
     <sb:ENUM INDEX="1" LABEL="Channel 1 out of range high"/>
     <sb:ENUM INDEX="2" LABEL="Channel 1 out of range low"/>
     <sb:ENUM INDEX="3" LABEL="Initialization Error (Channel 1)"/>
     <sb:ENUM INDEX="4" LABEL="Channel 2 out of range high"/>
     <sb:ENUM INDEX="5" LABEL="Channel 2 out of range low"/>
     <sb:ENUM INDEX="6" LABEL="Initialization Error (Channel 2)"/>
     <sb:ENUM INDEX="7" LABEL="Channel 1 and 2 Rationality Error"/>
     <sb:ENUM INDEX="1025" LABEL="Slow Channel Temperature out of range high"/>
     <sb:ENUM INDEX="1026" LABEL="Slow Channel Temperature out of range low"/>
     <sb:ENUM INDEX="1027" LABEL="Slow Channel Temperature initialization error"/>
     <sb:ENUM INDEX="1028" LABEL="Slow Channel Humidity out of range high"/>
     <sb:ENUM INDEX="1029" LABEL="Slow Channel Humidity out of range low"/>
     <sb:ENUM INDEX="1030" LABEL="Slow Channel Humidity initialization error"/>
     <sb:ENUM INDEX="1031" LABEL="Slow Channel Barometric Pressure out of range high"/>
```

```
    <sb:ENUM INDEX="1032" LABEL="Slow Channel Barometric Pressure out of range low"/>
    <sb:ENUM INDEX="1033" LABEL="Slow Channel Barometric Pressure initialization error"/>
   </sb:ENUM-VALUES>
  </sb:SIGNAL>
 </sb:SIGNALS>
</sb:FRAME>
<sb:FRAME NAME="Simu-Dual Throttle Position" STATE="ON">
 <sb:SHORT-NAME>DTP</sb:SHORT-NAME>
 <sb:DATA-SIZE>20</sb:DATA-SIZE>
 <sb:SIGNALS>
  <sb:SIGNAL ID="Channel_1">
   <sb:SHORT-NAME>TPS1</sb:SHORT-NAME>
   <sb:DESCRIPTION>""</sb:DESCRIPTION>
   <sb:BIT-POSITION>19</sb:BIT-POSITION>
   <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
   <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
   <sb:VALUE-TYPE>UNSIGNED_INT</sb:VALUE-TYPE>
   <sb:FACTOR>1.0</sb:FACTOR>
   <sb:OFFSET>0.0</sb:OFFSET>
   <sb:MIN>0</sb:MIN>
   <sb:MAX>10000.0</sb:MAX>
   <sb:UNIT></sb:UNIT>
  </sb:SIGNAL>
  <sb:SIGNAL ID="Channel_2">
   <sb:SHORT-NAME>TPS2</sb:SHORT-NAME>
   <sb:DESCRIPTION>""</sb:DESCRIPTION>
   <sb:BIT-POSITION>7</sb:BIT-POSITION>
   <sb:BIT-LENGTH>8</sb:BIT-LENGTH>
   <sb:BYTE-ORDER>LSB</sb:BYTE-ORDER>
   <sb:FACTOR>1.0</sb:FACTOR>
   <sb:OFFSET>0.0</sb:OFFSET>
   <sb:MIN>0</sb:MIN>
   <sb:MAX>10000.0</sb:MAX>
   <sb:UNIT></sb:UNIT>
  </sb:SIGNAL>
 </sb:SIGNALS>
</sb:FRAME>
<sb:FRAME NAME="Dual Throttle Position" STATE="OFF">
 <sb:SHORT-NAME>DTP</sb:SHORT-NAME>
 <sb:DATA-SIZE>24</sb:DATA-SIZE>
 <sb:SIGNALS>
  <sb:SIGNAL ID="Channel_1">
   <sb:SHORT-NAME>TPS1</sb:SHORT-NAME>
   <sb:DESCRIPTION>""</sb:DESCRIPTION>
   <sb:BIT-POSITION>23</sb:BIT-POSITION>
   <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
   <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
   <sb:VALUE-TYPE>UNSIGNED_INT</sb:VALUE-TYPE>
   <sb:FACTOR>1.0</sb:FACTOR>
   <sb:OFFSET>0.0</sb:OFFSET>
```

```
    <sb:MIN>0</sb:MIN>
    <sb:MAX>10000.0</sb:MAX>
    <sb:UNIT></sb:UNIT>
   </sb:SIGNAL>
   <sb:SIGNAL ID="Channel_2">
    <sb:SHORT-NAME>TPS2</sb:SHORT-NAME>
    <sb:DESCRIPTION>""</sb:DESCRIPTION>
    <sb:BIT-POSITION>11</sb:BIT-POSITION>
    <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
    <sb:BYTE-ORDER>LSB</sb:BYTE-ORDER>
    <sb:FACTOR>1.0</sb:FACTOR>
    <sb:OFFSET>0.0</sb:OFFSET>
    <sb:MIN>0</sb:MIN>
    <sb:MAX>10000.0</sb:MAX>
    <sb:UNIT></sb:UNIT>
   </sb:SIGNAL>
  </sb:SIGNALS>
 </sb:FRAME>
 <sb:FRAME NAME="Mass Air Flow (16)" STATE="OFF">
  <sb:SHORT-NAME>MAF/P</sb:SHORT-NAME>
  <sb:DATA-SIZE>24</sb:DATA-SIZE>
  <sb:SIGNALS>
   <sb:SIGNAL ID="Channel_1">
    <sb:SHORT-NAME>MAF</sb:SHORT-NAME>
    <sb:DESCRIPTION>""</sb:DESCRIPTION>
    <sb:BIT-POSITION>23</sb:BIT-POSITION>
    <sb:BIT-LENGTH>16</sb:BIT-LENGTH>
    <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
    <sb:FACTOR>1.0</sb:FACTOR>
    <sb:OFFSET>0.0</sb:OFFSET>
    <sb:MIN>0</sb:MIN>
    <sb:MAX>10000.0</sb:MAX>
    <sb:UNIT></sb:UNIT>
   </sb:SIGNAL>
  </sb:SIGNALS>
 </sb:FRAME>
 <sb:FRAME NAME="Mass Air Flow (16/8)" STATE="OFF">
  <sb:SHORT-NAME>MAF/P</sb:SHORT-NAME>
  <sb:DATA-SIZE>24</sb:DATA-SIZE>
  <sb:SIGNALS>
   <sb:SIGNAL ID="Channel_1">
    <sb:SHORT-NAME>MAF</sb:SHORT-NAME>
    <sb:DESCRIPTION>""</sb:DESCRIPTION>
    <sb:BIT-POSITION>23</sb:BIT-POSITION>
    <sb:BIT-LENGTH>16</sb:BIT-LENGTH>
    <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
    <sb:FACTOR>1.0</sb:FACTOR>
    <sb:OFFSET>0.0</sb:OFFSET>
    <sb:MIN>0</sb:MIN>
    <sb:MAX>10000.0</sb:MAX>
```

```
    <sb:UNIT></sb:UNIT>
   </sb:SIGNAL>
   <sb:SIGNAL ID="Channel_2">
    <sb:SHORT-NAME>Pressure</sb:SHORT-NAME>
    <sb:DESCRIPTION>""</sb:DESCRIPTION>
    <sb:BIT-POSITION>7</sb:BIT-POSITION>
    <sb:BIT-LENGTH>8</sb:BIT-LENGTH>
    <sb:BYTE-ORDER>LSB</sb:BYTE-ORDER>
    <sb:FACTOR>1.0</sb:FACTOR>
    <sb:OFFSET>0.0</sb:OFFSET>
    <sb:MIN>0</sb:MIN>
    <sb:MAX>10000.0</sb:MAX>
    <sb:UNIT></sb:UNIT>
   </sb:SIGNAL>
  </sb:SIGNALS>
 </sb:FRAME>
 <sb:FRAME NAME="Mass Air Flow (14/10)" STATE="OFF">
  <sb:SHORT-NAME>MAF/P</sb:SHORT-NAME>
  <sb:DATA-SIZE>24</sb:DATA-SIZE>
  <sb:SIGNALS>
   <sb:SIGNAL ID="Channel_1">
    <sb:SHORT-NAME>MAF</sb:SHORT-NAME>
    <sb:DESCRIPTION>""</sb:DESCRIPTION>
    <sb:BIT-POSITION>23</sb:BIT-POSITION>
    <sb:BIT-LENGTH>14</sb:BIT-LENGTH>
    <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
    <sb:FACTOR>1.0</sb:FACTOR>
    <sb:OFFSET>0.0</sb:OFFSET>
    <sb:MIN>0</sb:MIN>
    <sb:MAX>10000.0</sb:MAX>
    <sb:UNIT></sb:UNIT>
   </sb:SIGNAL>
   <sb:SIGNAL ID="Channel_2">
    <sb:SHORT-NAME>Pressure</sb:SHORT-NAME>
    <sb:DESCRIPTION>""</sb:DESCRIPTION>
    <sb:BIT-POSITION>9</sb:BIT-POSITION>
    <sb:BIT-LENGTH>10</sb:BIT-LENGTH>
    <sb:BYTE-ORDER>LSB</sb:BYTE-ORDER>
    <sb:FACTOR>1.0</sb:FACTOR>
    <sb:OFFSET>0.0</sb:OFFSET>
    <sb:MIN>0</sb:MIN>
    <sb:MAX>10000.0</sb:MAX>
    <sb:UNIT></sb:UNIT>
   </sb:SIGNAL>
  </sb:SIGNALS>
 </sb:FRAME>
 <sb:FRAME NAME="Single Secure Sensor" STATE="OFF">
  <sb:SHORT-NAME>SSS</sb:SHORT-NAME>
  <sb:DATA-SIZE>24</sb:DATA-SIZE>
  <sb:SIGNALS>
```

```
  <sb:SIGNAL ID="Channel_1">
   <sb:SHORT-NAME>Ch1</sb:SHORT-NAME>
   <sb:DESCRIPTION>""</sb:DESCRIPTION>
   <sb:BIT-POSITION>23</sb:BIT-POSITION>
   <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
   <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
   <sb:FACTOR>1.0</sb:FACTOR>
   <sb:OFFSET>0.0</sb:OFFSET>
   <sb:MIN>0</sb:MIN>
   <sb:MAX>10000.0</sb:MAX>
   <sb:UNIT></sb:UNIT>
  </sb:SIGNAL>
  <sb:SIGNAL ID="Channel_2">
   <sb:SHORT-NAME>Counter</sb:SHORT-NAME>
   <sb:DESCRIPTION>""</sb:DESCRIPTION>
   <sb:BIT-POSITION>11</sb:BIT-POSITION>
   <sb:BIT-LENGTH>8</sb:BIT-LENGTH>
   <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
   <sb:FACTOR>1.0</sb:FACTOR>
   <sb:OFFSET>0.0</sb:OFFSET>
   <sb:MIN>0</sb:MIN>
   <sb:MAX>256.0</sb:MAX>
   <sb:UNIT></sb:UNIT>
  </sb:SIGNAL>
 </sb:SIGNALS>
</sb:FRAME>
<sb:FRAME NAME="Pressure Sensor" STATE="OFF">
 <sb:SHORT-NAME>P</sb:SHORT-NAME>
 <sb:DATA-SIZE>24</sb:DATA-SIZE>
 <sb:SIGNALS>
  <sb:SIGNAL ID="Channel_1">
   <sb:SHORT-NAME>Pressure1</sb:SHORT-NAME>
   <sb:DESCRIPTION>""</sb:DESCRIPTION>
   <sb:BIT-POSITION>23</sb:BIT-POSITION>
   <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
   <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
   <sb:FACTOR>1.0</sb:FACTOR>
   <sb:OFFSET>0.0</sb:OFFSET>
   <sb:MIN>0</sb:MIN>
   <sb:MAX>10000.0</sb:MAX>
   <sb:UNIT></sb:UNIT>
  </sb:SIGNAL>
  <sb:SIGNAL ID="Channel_2">
   <sb:SHORT-NAME>Pressure2</sb:SHORT-NAME>
   <sb:DESCRIPTION>""</sb:DESCRIPTION>
   <sb:BIT-POSITION>11</sb:BIT-POSITION>
   <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
   <sb:BYTE-ORDER>LSB</sb:BYTE-ORDER>
   <sb:FACTOR>1.0</sb:FACTOR>
   <sb:OFFSET>0.0</sb:OFFSET>
```

```
    <sb:MIN>0</sb:MIN>
    <sb:MAX>10000.0</sb:MAX>
    <sb:UNIT></sb:UNIT>
   </sb:SIGNAL>
  </sb:SIGNALS>
 </sb:FRAME>
 <sb:FRAME NAME="Pressure and Temperature Sensor" STATE="OFF">
  <sb:SHORT-NAME>P/T</sb:SHORT-NAME>
  <sb:DATA-SIZE>24</sb:DATA-SIZE>
  <sb:SIGNALS>
   <sb:SIGNAL ID="Channel_1">
    <sb:SHORT-NAME>Pressure</sb:SHORT-NAME>
    <sb:DESCRIPTION>""</sb:DESCRIPTION>
    <sb:BIT-POSITION>23</sb:BIT-POSITION>
    <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
    <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
    <sb:FACTOR>1.0</sb:FACTOR>
    <sb:OFFSET>0.0</sb:OFFSET>
    <sb:MIN>0</sb:MIN>
    <sb:MAX>10000.0</sb:MAX>
    <sb:UNIT></sb:UNIT>
   </sb:SIGNAL>
   <sb:SIGNAL ID="Channel_2">
    <sb:SHORT-NAME>Temperature</sb:SHORT-NAME>
    <sb:DESCRIPTION>""</sb:DESCRIPTION>
    <sb:BIT-POSITION>11</sb:BIT-POSITION>
    <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
    <sb:BYTE-ORDER>LSB</sb:BYTE-ORDER>
    <sb:FACTOR>1.0</sb:FACTOR>
    <sb:OFFSET>0.0</sb:OFFSET>
    <sb:MIN>0</sb:MIN>
    <sb:MAX>10000.0</sb:MAX>
    <sb:UNIT></sb:UNIT>
   </sb:SIGNAL>
  </sb:SIGNALS>
 </sb:FRAME>
 <sb:FRAME NAME="Pressure and Secure Sensor" STATE="OFF">
  <sb:SHORT-NAME>P/S</sb:SHORT-NAME>
  <sb:DATA-SIZE>24</sb:DATA-SIZE>
  <sb:SIGNALS>
   <sb:SIGNAL ID="Channel_1">
    <sb:SHORT-NAME>Pressure</sb:SHORT-NAME>
    <sb:DESCRIPTION>""</sb:DESCRIPTION>
    <sb:BIT-POSITION>23</sb:BIT-POSITION>
    <sb:BIT-LENGTH>12</sb:BIT-LENGTH>
    <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
    <sb:FACTOR>1.0</sb:FACTOR>
    <sb:OFFSET>0.0</sb:OFFSET>
    <sb:MIN>0</sb:MIN>
    <sb:MAX>10000.0</sb:MAX>
```

```
      <sb:UNIT></sb:UNIT>
     </sb:SIGNAL>
     <sb:SIGNAL ID="Channel_2">
      <sb:SHORT-NAME>Counter</sb:SHORT-NAME>
      <sb:DESCRIPTION>""</sb:DESCRIPTION>
      <sb:BIT-POSITION>11</sb:BIT-POSITION>
      <sb:BIT-LENGTH>8</sb:BIT-LENGTH>
      <sb:BYTE-ORDER>MSB</sb:BYTE-ORDER>
      <sb:FACTOR>1.0</sb:FACTOR>
      <sb:OFFSET>0.0</sb:OFFSET>
      <sb:MIN>0</sb:MIN>
      <sb:MAX>10000.0</sb:MAX>
      <sb:UNIT></sb:UNIT>
     </sb:SIGNAL>
    </sb:SIGNALS>
   </sb:FRAME>
  </sb:FRAMES>
</sb:LABEL-LIST-FILE>
```


13.10.5 SENT decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".

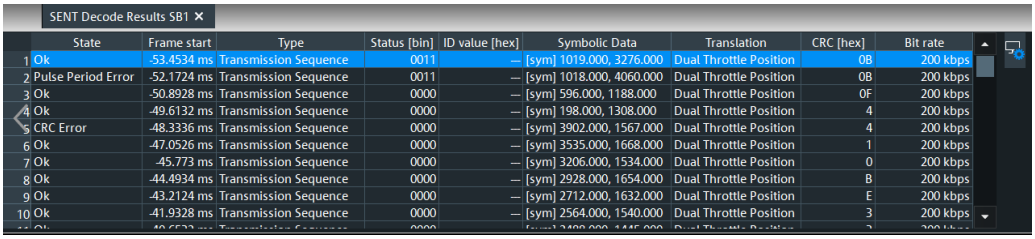
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Data is decoded and displayed in the order of its reception. The "Decode results" box shows the detailed decoded data for each frame as it is received.



SENT Decode Results SB1

	State	Frame start	Type	Status [bin]	ID value [hex]	Symbolic Data	Translation	CRC [hex]	Bit rate
1	Ok	-53.4534 ms	Transmission Sequence	0011	—	[sym] 1019.000, 3276.000	Dual Throttle Position	0B	200 kbps
2	Pulse Period Error	-52.1724 ms	Transmission Sequence	0011	—	[sym] 1018.000, 4060.000	Dual Throttle Position	0B	200 kbps
3	Ok	-50.8928 ms	Transmission Sequence	0000	—	[sym] 596.000, 1188.000	Dual Throttle Position	0F	200 kbps
4	Ok	-49.6132 ms	Transmission Sequence	0000	—	[sym] 198.000, 1308.000	Dual Throttle Position	4	200 kbps
5	CRC Error	-48.3336 ms	Transmission Sequence	0000	—	[sym] 3902.000, 1567.000	Dual Throttle Position	4	200 kbps
6	Ok	-47.0526 ms	Transmission Sequence	0000	—	[sym] 3535.000, 1668.000	Dual Throttle Position	1	200 kbps
7	Ok	-45.773 ms	Transmission Sequence	0000	—	[sym] 3206.000, 1534.000	Dual Throttle Position	0	200 kbps
8	Ok	-44.4934 ms	Transmission Sequence	0000	—	[sym] 2928.000, 1654.000	Dual Throttle Position	B	200 kbps
9	Ok	-43.2124 ms	Transmission Sequence	0000	—	[sym] 2712.000, 1632.000	Dual Throttle Position	E	200 kbps
10	Ok	-41.9328 ms	Transmission Sequence	0000	—	[sym] 2564.000, 1540.000	Dual Throttle Position	3	200 kbps

Table 13-15: Content of the decode result table

Column	Description
State	Overall state of the frame
Frame start	Time of frame start in relation to the trigger point
Type	Frame type
Status [bin]	Status value
ID Value [hex]	Identifier value
Symbolic Data	Symbolic data value
Translation	Symbolic label name defined in the label list
CRC [hex]	CRC sequence value
Bit rate	Value of the bit rate

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands to retrieve decode results are described in Chapter 24.17.11.3, "Decode results", on page 1909.

13.10.6 Search on decoded SENT data

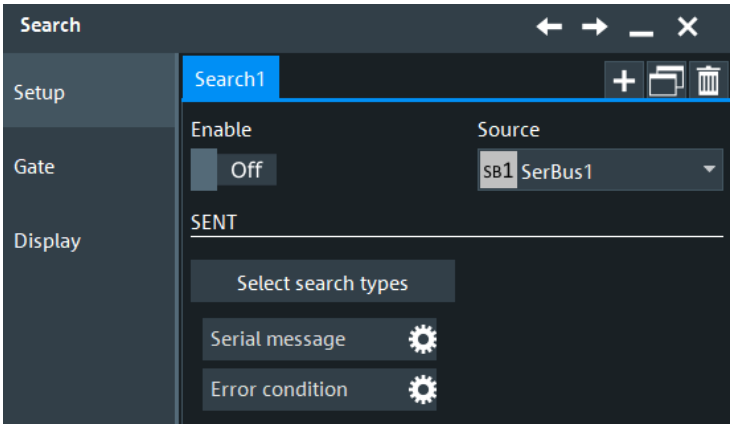
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 11, "Search functions", on page 419.

13.10.6.1 SENT search setup

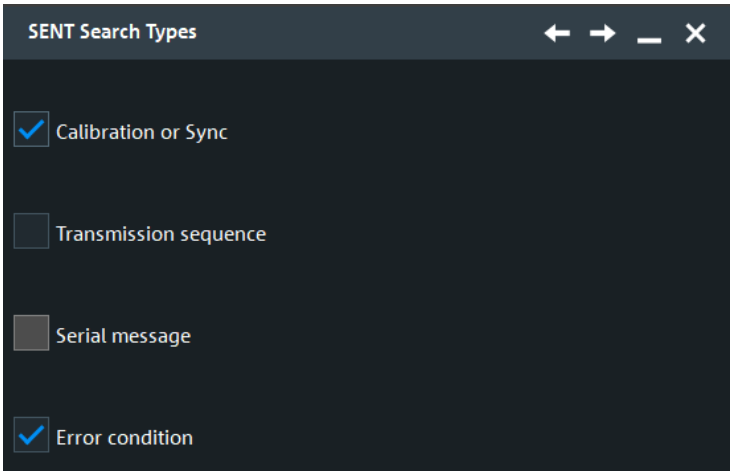
Access: [Bus] > "Shortcuts" tab > "Setup search"



Search types

Enable the events to be searched for.

Unlike triggering, where you can trigger only on one defined event, you can search for various different events in one search.



"Calibration Sync"

Searches for the end of the "Calibration/Synchronization" sequence, i.e. the falling edge.
This setting does not require any input parameters.

"Transmission Seq"

Searches for the end of the status nibble in a single transmission sequence, or the end of the combination of the status and data nibble(s).
Description of the specific settings: "Transmission Sequence setup" on page 662

"Serial Message"

Searching on a serial message depends on the serial protocol selected with Serial protocol > Short | Enhanced and the associated setting parameters:

- "Sequence > Identifier": searches for the end of the identifier nibble.
- "Sequence > ID+Data": searches for the end of the "ID and Data" nibble.

Description of the serial messages specific settings: "Serial Message setup" on page 663

"Error Condition"

Searches for the end of certain error events.
Description of trigger type specific settings: "Error conditions setup" on page 664

Remote command:
`SEARch:TRIGger:SENT:CALibration` on page 1917
`SEARch:TRIGger:SENT:TRANsmission` on page 1917
`SEARch:TRIGger:SENT:SMSG` on page 1917
`SEARch:TRIGger:SENT:ERRor` on page 1918

Transmission Sequence setup
Configures the search conditions for the transmission sequence.

Note: The displayed parameters depend on the selected "Sequence". The instrument displays the data setting parameters when you select "Status+Data", see "Sequence" on page 662.
The search type specific conditions are the same as for the trigger type, see "Transmission Sequence setup" on page 645.

Sequence ← Transmission Sequence setup
Selects the condition for searching in a single transmission sequence.

You can search for the end of the status nibble or the combination of the status and data nibble(s).

Remote command:
`SEARch:TRIGger:SENT:TTYPe` on page 1918

Status ← Transmission Sequence setup
Defines the data bits for the status nibble.

Remote command:
`SEARch:TRIGger:SENT:STATus` on page 1919

Condition ← Transmission Sequence setup
Selects the operator to define a specific data pattern or a data range.

The available operators:
- Equal, Not equal
- Less than, Greater than
- Less or equal, Greater or equal
- In range, Out of range

Remote command:
`SEARch:TRIGger:SENT:TDCN` on page 1919

Data Nibbles ← Transmission Sequence setup
Sets the number of data units in a single transmission sequence.

The maximum number of data nibbles is 6.

Remote command:
`BUS<m>:SENT:DNIBbles` on page 1901

Data min ← Transmission Sequence setup
Sets the data pattern. Enter the bytes in msb first bit order. The maximum pattern length is 64 bit. Waveform data is compared with the pattern byte-by-byte.

Remote command:
`SEARch:TRIGger:SENT:TDMN` on page 1919

Data max ← Transmission Sequence setup
The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:SENT:TDMX` on page 1920

Serial Message setup
Configures the search conditions for a serial message.

Note: The displayed parameters depend on the selected "Sequence". The instrument indicates the data setting parameters when you select "ID+Data", see "Sequence" on page 663.
The description of the search type specific settings are the same as for the trigger type, see "Serial Message setup" on page 646.

Sequence ← Serial Message setup
Selects the condition for searching in a serial message.

You can search for the end of an identifier nibble or the combination of the identifier and data nibble(s).

Remote command:
`SEARch:TRIGger:SENT:STYPe` on page 1920

ID Type ← Serial Message setup
Selects the message ID format for the enhanced serial message type.

You can select either 4 bit or 8 bit message ID.

Remote command:
`SEARch:TRIGger:SENT:SIDType` on page 1920

Identifier Condition ← Serial Message setup
Selects the operator to set a specific identifier or an identifier range.

The available operators:
- Equal, Not equal

- Less than, Greater than
- Less or equal, Greater or equal
- In range, Out of range

Remote command:
`SEARch:TRIGger:SENT:SICN` on page 1921

Identifier min ← Serial Message setup
Defines the bit pattern of the message identifier.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

Remote command:
`SEARch:TRIGger:SENT:SIMN` on page 1921

Identifier max ← Serial Message setup
The second identifier pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:SENT:SIMX` on page 1921

SSM Data Condition ← Serial Message setup
Selects the operator to set a specific data pattern or a data range.

The available operators:
- Equal, Not equal
- Less than, Greater than
- Less or equal, Greater or equal
- In range, Out of range

Remote command:
`SEARch:TRIGger:SENT:SDCN` on page 1922

Data min ← Serial Message setup
Sets the data pattern. Enter the bytes in msb first bit order. The maximum pattern length is 64 bit. Waveform data is compared with the pattern byte-by-byte.

Remote command:
`SEARch:TRIGger:SENT:SDMN` on page 1922

Data max ← Serial Message setup
The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:SENT:SDMX` on page 1922

Error conditions setup
Performs the search on one or more of the following error events:
- "Transmission sequence errors"
  - "Calibration Pulse Error"
    Searches for calibration pulse errors.

  - "Pulse Period Error"
    Searches for pulse period errors in a transmission sequence.
  - "Irregular Frame Length Error"
    Searches for irregular frame length errors in a transmission sequence if pause pulse mode is set to constant frame length..
- Serial message error
  - "Form Error"
    Searches for format errors in serial messages.
- "CRC Error"
  Searches for errors in the complete data transmission.

Remote command:
`SEARch:TRIGger:SENT:PULSeerror` on page 1923
`SEARch:TRIGger:SENT:PPERioderror` on page 1923
`SEARch:TRIGger:SENT:FORMerror` on page 1923
`SEARch:TRIGger:SENT:CRCerror` on page 1924
`SEARch:TRIGger:SENT:IRFLength` on page 1923

13.10.6.2 SENT search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands:

- `SEARch:RESult:SENT:FCOunt?` on page 1925
- `SEARch:RESult:SENT:FRAMe<m>:STATus?` on page 1928
- `SEARch:RESult:SENT:FRAMe<m>:STARt?` on page 1928
- `SEARch:RESult:SENT:FRAMe<m>:STOP?` on page 1928
- `SEARch:RESult:SENT:FRAMe<m>:DATA?` on page 1925
- `SEARch:RESult:SENT:FRAMe<m>:CSValue?` on page 1925
- `SEARch:RESult:SENT:FRAMe<m>:IDTYpe?` on page 1925
- `SEARch:RESult:SENT:FRAMe<m>:IDValue?` on page 1926
- `SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:STATe?` on page 1926
- `SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:VALue?` on page 1926
- `SEARch:RESult:SENT:FRAMe<m>:PAPTicks?` on page 1927
- `SEARch:RESult:SENT:FRAMe<m>:SCOM?` on page 1927
- `SEARch:RESult:SENT:FRAMe<m>:SDATa?` on page 1927
- `SEARch:RESult:SENT:FRAMe<m>:SYMBol?` on page 1928
- `SEARch:RESult:SENT:FRAMe<m>:SYNCduration?` on page 1929

- `SEARch:RESult:SENT:FRAMe<m>:TYPE?` on page 1929

13.11 CXPI (option R&S RTO6-K520)

The Clock Extension Peripheral Interface (CXPI) protocol defines a communication standard for the vehicles electric system. It is developed by the Society of Automotive Engineers of Japan, Inc. (JSAE).

13.11.1 The CXPI protocol

This chapter provides an overview of the protocol characteristics, frame types and frame fields.

CXPI characteristics

Main characteristics of CXPI are:

- Carrier Sense Multiple Access (CSMA) and Collision Resolution (CR)
- The master sends pulse width modulated clock (PWMC)
- Support of the sleep/wake function to reduce power consumption
- Support of two methods for frame transfer management, the event trigger method and the polling method.
  The event trigger method can be used when the focus is placed on the responsiveness of the slave node communication. The polling method puts the focus on the communication periodicity.
- Maximum baud rate of 20kbit/s
- Up to 16 nodes connected to a communication bus

Frame types

The CXPI protocol defines three types of frames:

- **Normal frame**: it varies in length according to the frame transfer method. The PTYPE field is only used for the polling method.
  The normal frame is used to transfer compact data.

Figure 13-19: CXPI normal frame

- **Sleep frame**: it has a fixed length and fixed values for the PID and data fields. It is sent by the master node to command the slave nods to change into sleep state.

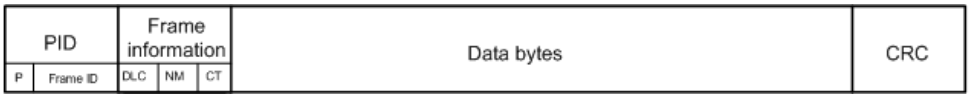


Figure 13-20: CXPI sleep frame

- **Long frame**: it varies in length according to the frame transfer method. The PTYPE field is only used for the polling method.
  The long frame is used to transfer large amount of data.

PTYPE		PID		Frame information				Data bytes	CRC
P	Frame Type	P	Frame ID	DLC	NM	CT	extension DLC		

Figure 13-21: CXPI long frame

During measurements further type of frames may also occur: PTYPE only, PID only and PTYPE + PID only.

Frame fields

The different frames may contain the following fields:

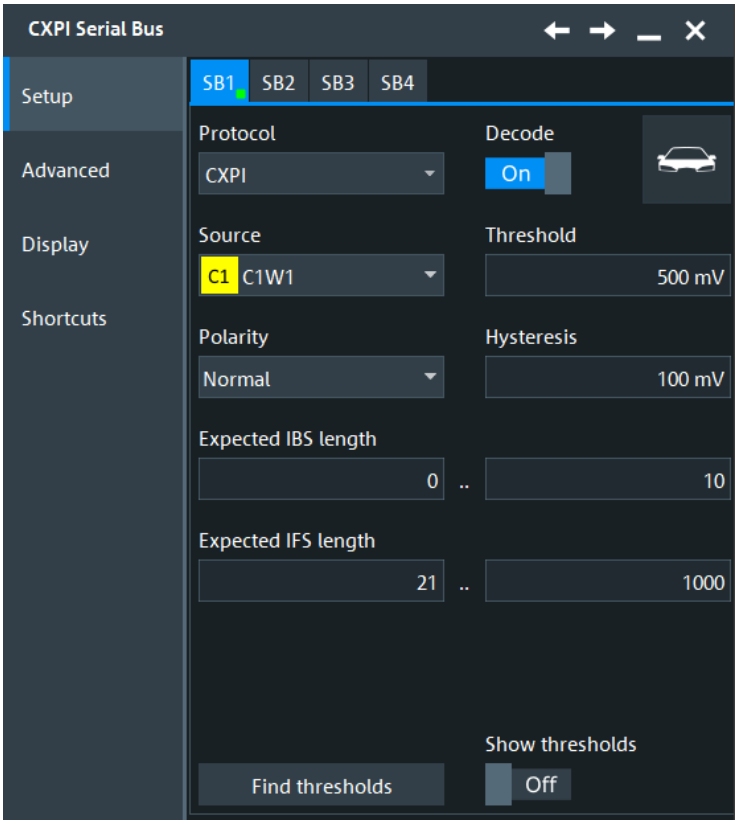
- **PID**: consists of a frame identifier (7 bits) and one parity bit.
- **Protected TYPE (PTYPE)**: a special PID field with a frame identifier of 00 (hex) and a parity of 1. It is used only in the polling method.
- **Frame information**: it consists of the following fields:
  - **Data Length Code (DLC)**: 4 bits that indicate the length of the data byte of a normal frame.
  - **Network Management (NM)**: 2 bits indicating if the frame is in wakeup or sleep mode.
  - **Counter (CT)**: 2 bits indicating the continuity of the frame.
  - **extension DLC**: 1 byte that is added to the frame information for long frames. It indicates the length of the data byte field of a long frame.
- **Data byte**: the length of the data byte field depends on the frame type. It contains the actual information of the frame.
- **Cyclic Redundancy Check (CRC)**: 1 byte for normal and sleep frames and 2 bytes for long frames. CRC errors can be detected to check if the received data is correct.

13.11.2 CXPI configuration

13.11.2.1 CXPI configuration settings

Access: [Bus] key > "Setup" tab > "Protocol" = "CXPI"

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Source
Sets the source channel for the signal.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Alternatively, digital channels can be used if MSO option R&S RTO6-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Remote command:
`BUS<m>:CXPI:SDATa` on page 1932

Polarity
Selects if the polarity of the transmitted waveform is normal (high = 1) or inverted (low = 1).

Remote command:
`BUS<m>:CXPI:POLarity` on page 1931

Threshold
Sets the threshold value for digitization of the signal. If the signal value on the line is higher than the threshold, the signal state is high. If the signal value on the line is below the threshold, the signal state is low. The interpretation of high and low is defined by the Polarity.

There are two ways to set the threshold:

"Threshold"	Enter the value directly in the field.
"Find thresholds"	Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
`BUS<m>:CXPI:THReshold` on page 1931
`BUS<m>:SETReflevels` on page 1671

Hysteresis
Sets a value for the hysteresis.

The hysteresis is applied as ("Threshold" +"Hysteresis") and ("Threshold" - "Hysteresis").

Remote command:
`BUS<m>:CXPI:HYSTeresis` on page 1931

Expected IBS length
Sets a range for the expected inter-byte-space (IBS) length.

The standard has a permissible IBS length. Violation of these limits leads to decoding ambiguities. To test different limits, you can set the expected IBS length value.

Remote command:
`BUS<m>:CXPI:IBS:MAX` on page 1932
`BUS<m>:CXPI:IBS:MIN` on page 1932

Expected IFS length
Sets a range for the inter-frame-space.

The standard has a permissible IFS length. Violation of these limits leads to decoding ambiguities. To test different limits, you can set the expected IFS length value.

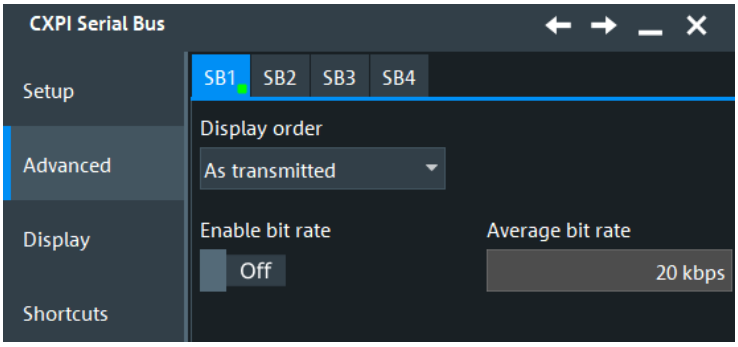
Remote command:
`BUS<m>:CXPI:IFS:MAX` on page 1932
`BUS<m>:CXPI:IFS:MIN` on page 1933

Show thresholds
If enabled, the threshold levels are displayed in the diagram.

Remote command:
`BUS<m>:THReshold` on page 1671

13.11.2.2 Advanced settings

Access: [Bus] > "Advanced" tab.



Display order
Selects the order in which the signal is displayed in the honeycomb.

"As transmitted"
The signal is displayed in the order it occurs.

"Logical" The signal is displayed according to the definition of the standard (MSB bit order).

Remote command:
`BUS<m>:CXPI:DORD` on page 1931

Enable bit rate/ Average bit rate
If enabled, sets the number of transmitted bits per second. If disabled, the bit rate is measured and the average bit rate in the acquisition window is displayed.

Remote command:
`BUS<m>:CXPI:BITRate:ENABle` on page 1930
`BUS<m>:CXPI:BITRate:VALue` on page 1930
`BUS<m>:CXPI:RESult:BITRate?` on page 1930

13.11.2.3 Display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

13.11.2.4 Configuring the CXPI signals

For configuration, assign the lines to the input channels and define the logical thresholds and the hysteresis.

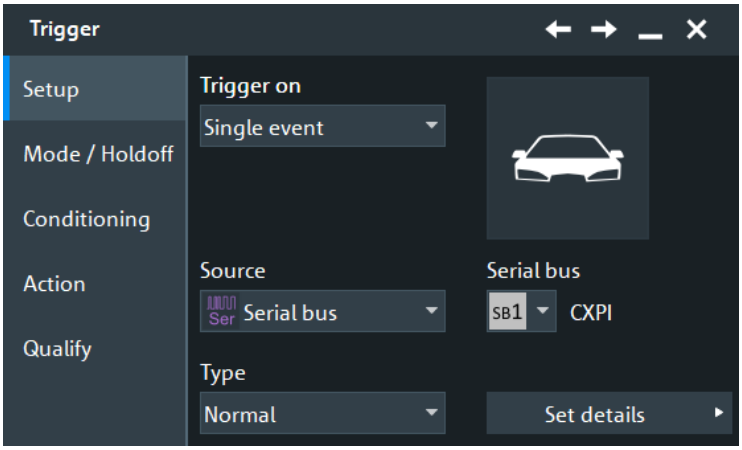
1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. Select the "Setup" tab.
4. Tap the "Protocol" button and select the protocol: "CXPI".
5. Tap the "Variant" button, and select the protocol.
6. Select the source and the polarity.

7. Enter the "Threshold" and the "Hysteresis".
8. Enable "Decode".

13.11.3 CXPI trigger

13.11.3.1 CXPI trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Trigger type
Selects the trigger type for CXPI analysis.

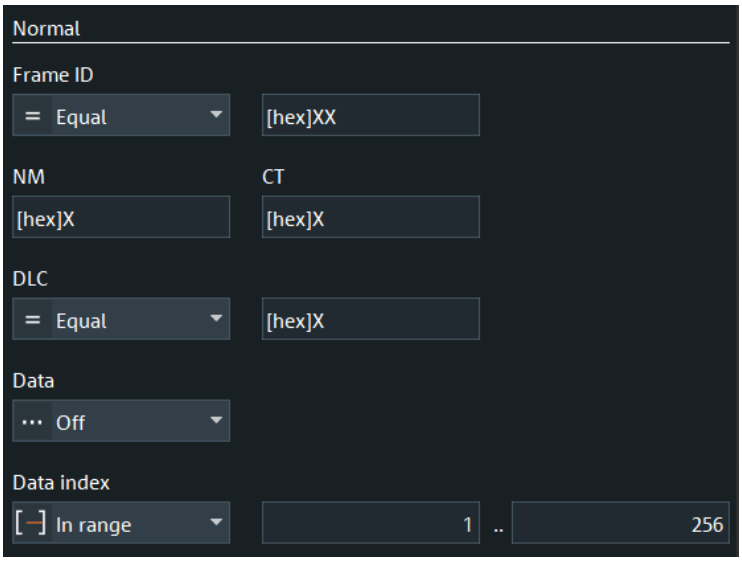
Remote command:
`TRIGger<m>:CXPI:TYPE` on page 1940

Frame Start ← Trigger type
Triggers on a frame start.

Normal ← Trigger type
Triggers on a normal frame.

You can refine the trigger condition:

- Set a frame ID pattern or a frame ID range to trigger only on frames that fulfill these conditions.
  See: "Frame ID" on page 676
- Set an NM value expected in the frame.
  See "NM" on page 678.
- Set a CT value expected in the frame.
  See: "CT" on page 678.
- Set a DLC pattern or DLC pattern range to trigger only on frames that fulfill these conditions.
  See: "DLC" on page 677.
- Set a data pattern condition to trigger only on frames that fulfill these conditions.
  See: "Data" on page 677.
- Set an index or index range to trigger only on frames that fulfill these conditions.
  See: "Index" on page 678.

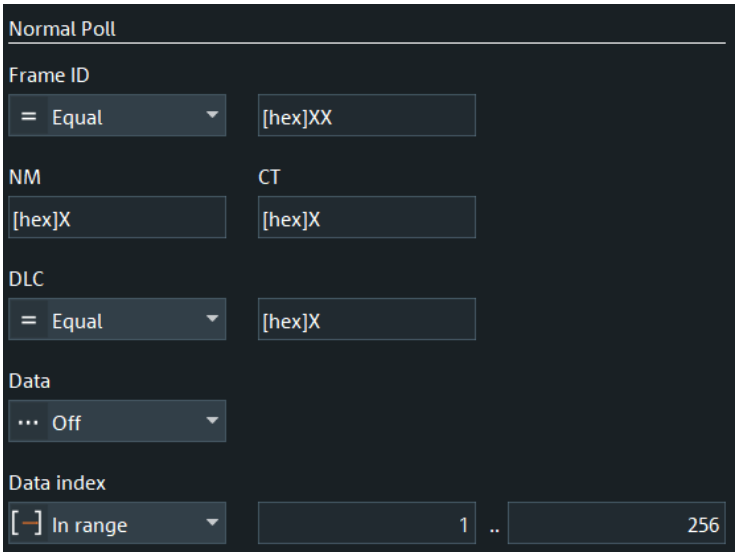


Normal Poll ← Trigger type
Triggers on a normal poll frame.

You can refine the trigger condition:

- Set a frame ID pattern or a frame ID range to trigger only on frames that fulfill these conditions.
  See: "Frame ID" on page 676
- Set an NM value expected in the frame.
  See "NM" on page 678.
- Set a CT value expected in the frame.
  See: "CT" on page 678.
- Set a DLC pattern or DLC pattern range to trigger only on frames that fulfill these conditions.
  See: "DLC" on page 677.
- Set a data pattern condition to trigger only on frames that fulfill these conditions.

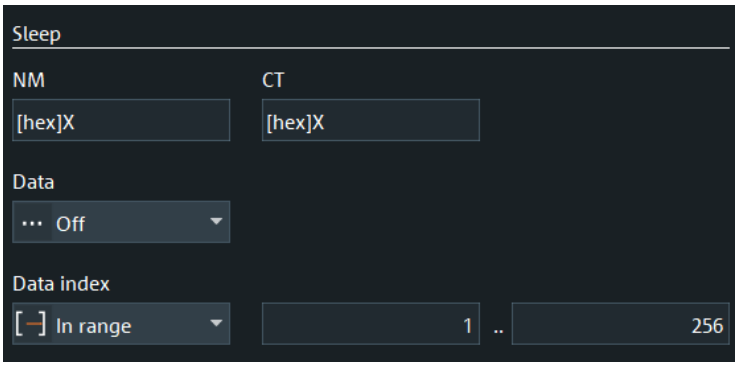
See: "Data" on page 677.
- Set an index or index range to trigger only on frames that fulfill these conditions.
  See: "Index" on page 678.



Sleep ← Trigger type
Triggers on a sleep frame.

You can refine the trigger condition:

- Set an NM value expected in the frame.
  See "NM" on page 678.
- Set a CT value expected in the frame.
  See: "CT" on page 678.
- Set a data pattern condition to trigger only on frames that fulfill these conditions.
  See: "Data" on page 677.
- Set an index or index range to trigger only on frames that fulfill these conditions.
  See: "Index" on page 678.

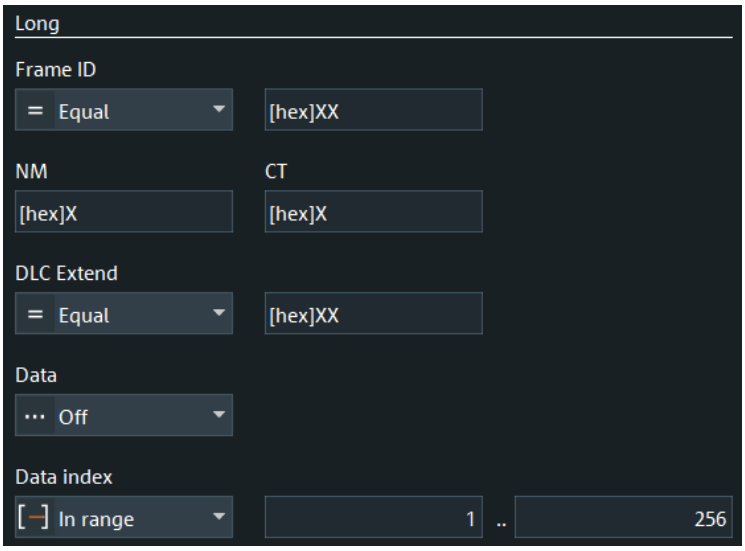


Long ← Trigger type
Triggers on a long frame.

You can refine the trigger condition:

- Set a frame ID pattern or a frame ID range to trigger only on frames that fulfill these conditions.

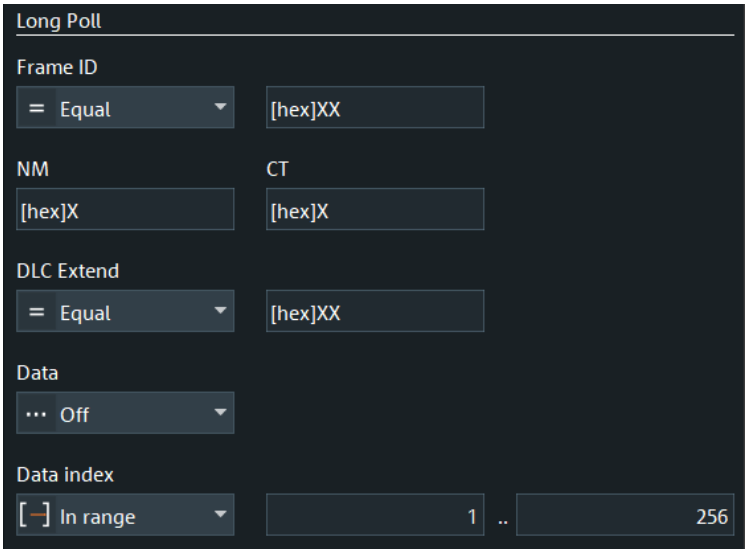
See: "Frame ID" on page 676
- Set an NM value expected in the frame.
  See "NM" on page 678.
- Set a CT value expected in the frame.
  See: "CT" on page 678.
- Set a DLC extend pattern or DLC extend pattern range to trigger only on frames that fulfill these conditions.
  See: "DLC Extend" on page 677.
- Set a data pattern condition to trigger only on frames that fulfill these conditions.
  See: "Data" on page 677.
- Set an index or index range to trigger only on frames that fulfill these conditions.
  See: "Index" on page 678.



Long Poll ← Trigger type
Triggers on a long poll frame.

You can refine the trigger condition:

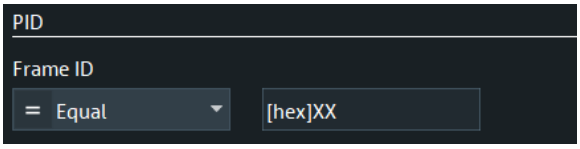
- Set a frame ID pattern or a frame ID range to trigger only on frames that fulfill these conditions.
  See: "Frame ID" on page 676
- Set an NM value expected in the frame.
  See "NM" on page 678.
- Set a CT value expected in the frame.
  See: "CT" on page 678.
- Set a DLC extend pattern or DLC extend pattern range to trigger only on frames that fulfill these conditions.
  See: "DLC Extend" on page 677.
- Set a data pattern condition to trigger only on frames that fulfill these conditions.
  See: "Data" on page 677.
- Set an index or index range to trigger only on frames that fulfill these conditions.
  See: "Index" on page 678.



PID ← Trigger type
Triggers on a protected ID (PID) field. A PID field consists of a frame identifier and a parity bit.

Optionally, you can define the frame ID pattern or a frame ID range to trigger only on frames that fulfill these conditions.

See:"Frame ID" on page 676.



PTYPE ← Trigger type
Triggers on a protected TYPE (PTYPE) field, a special PID field with a frame ID of 00 (hex) and a parity of 1. It is used only in the polling method.

PTYPE + PID ← Trigger type
Triggers on a PTYPE field followed by a PID field.

Optionally, you can define a frame ID pattern or a frame ID range to trigger only on frames that fulfill these conditions.

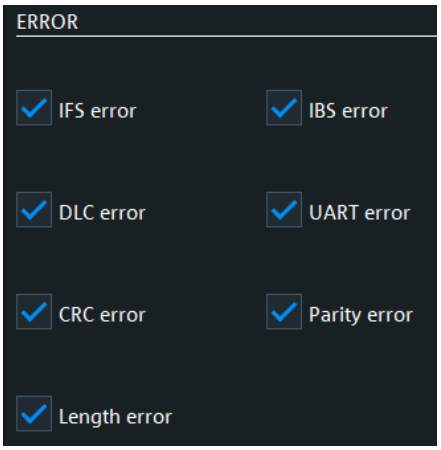
See:"Frame ID" on page 676.

Error ← Trigger type
Triggers if one or more of the following errors occur:

- "IFS error"
  The inter-frame space (IFS) is a bit field after a frame that has a logical value 1. It is used to separate two consecutive frames. IFS errors occur when the length of the IFS field differs from the one defined in the protocol.
- "IBS error"

The inter-byte-space (IBS) is the interval between each byte within a frame. IBS errors occur when the length of the IBS field differs from the one defined in the protocol.

- "Length error"
  Length error occurs when there are not enough bits/words to build a frame.
- "DLC error"
  Data length code error occurs when the value of the DLC field is different from the data byte field value. If there is a DLC error, it is still possible to build a frame, but there are too many/too few data words.
- "CRC error"
  The transmitting node calculates the cyclic redundancy check (CRC) value of a frame and stores it into the CRC field of the frame. CRC error occurs when this CRC field value differs from the value calculated by the receiving node.
- "Parity error"
  Triggers on a parity error indicating a transmission error.
- "UART error"
  For the CXPIstandard, each 8-bit byte is transported as a 10-bit UART word, framed by a start bit (1) and stop bit (0). An UART error occurs if the identified symbol does not follow the described pattern, if it is too short or too long.



Frame ID

The frame identifier is a 7-bit field. Its setup consists of the condition and one or two frame patterns.

"Condition"	Defines the operator to set a specific frame identifier ("Equal" or "Not equal") or a frame identifier range.
"Frame ID min / Frame ID"	Defines the bit pattern of the frame identifier. In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.
"Frame ID max"	The second frame identifier pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:

`TRIGger<m>:CXPI:FID:CONDition` on page 1939
`TRIGger<m>:CXPI:FID:MAX` on page 1940
`TRIGger<m>:CXPI:FID:MIN` on page 1940

DLC
Sets the data length code, which indicates the length of the data byte.

The DLC setup consists of the condition and one or two DLC patterns.

"Condition" Defines the operator to set a specific DLC ("Equal" or "Not equal") or a DLC range.

"DLC min / DLC " Defines the bit pattern of the DLC.
In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"DLC max" The second DLC pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:CXPI:DLC:CONDition` on page 1937
`TRIGger<m>:CXPI:DLC:MAX` on page 1937
`TRIGger<m>:CXPI:DLC:MIN` on page 1937

DLC Extend
Sets the extension data length code, which indicates the length of the data byte of a long frame.

The DLC extend setup consists of the condition and one or two DLC patterns.

"Condition" Defines the operator to set a specific DLC extend ("Equal" or "Not equal") or a DLC extend range.

"DLC Extend min / DLC Extend" Defines the bit pattern of the DLC extend.
In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"DLC Extend max" The second DLC extend pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:CXPI:DEXTension:CONDition` on page 1936
`TRIGger<m>:CXPI:DEXTension:MAX` on page 1936
`TRIGger<m>:CXPI:DEXTension:MIN` on page 1936

Data
The data setup consists of the condition and one or two identifier pattern.

"Condition" Defines the operator to set a specific data ("Equal" or "Not equal") or a data range.

"Data min / Data" Defines the bit pattern of the data.
In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:CXPI:DATA:DCONdition` on page 1934
`TRIGger<m>:CXPI:DATA:DMAX` on page 1934
`TRIGger<m>:CXPI:DATA:DMIN` on page 1935

Index
The data index setup consists of the condition and one or two index patterns.

"Condition"	Defines the operator to set a specific data index ("Equal" or "Not equal") or data index range.
"Index min / Index"	Defines the bit pattern of the data index.
"Index max"	The second data index is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:CXPI:DATA:ICONdition` on page 1935
`TRIGger<m>:CXPI:DATA:IMAX` on page 1935
`TRIGger<m>:CXPI:DATA:IMIN` on page 1935

NM
Sets the value of the network management (NM) field, 2 bits indicating if the frame is in wakeup or sleep mode.

Remote command:
`SEARch:TRIGger:CXPI:NM` on page 1952

CT
Sets the value of the counter (CT), 2 bits indicating the continuity of the frame.

Remote command:
`TRIGger<m>:CXPI:CT` on page 1934

13.11.3.2 Triggering on CXPI

Prerequisite: A bus is configured for the CXPI signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [Bus] key and select the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to CXPI.
5. Tap "Trigger Type" and select the trigger type to be used for CXPI protocol analysis.
6. Depending on the selected trigger type, more setup conditions can be specified.

For information on how to proceed with the configuration settings, see Chapter 13.11.3.1, "CXPI trigger settings", on page 671.

13.11.4 Search on decoded CXPI data

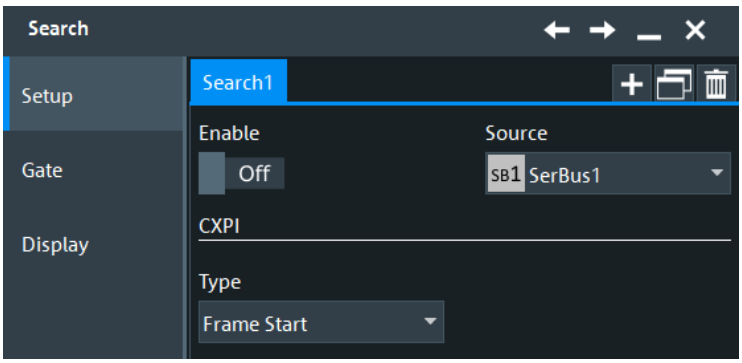
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 11, "Search functions", on page 419.

13.11.4.1 Search settings

Access: [Bus] > "Shortcuts" tab > "Setup search"



Search type
Selects the condition type for the CXPI search.

"Frame Start" Searches for a frame start.

"Normal" Searches for a normal frame.
You can refine the search condition, see:

- "Frame ID" on page 680
- "NM" on page 681
- "CT" on page 681
- "DLC" on page 680
- "Data" on page 681
- "Index" on page 681

"Normal Poll" Searches for a normal poll frame.
You can refine the search condition, see:

- "Frame ID" on page 680
- "NM" on page 681
- "CT" on page 681
- "DLC" on page 680
- "Data" on page 681
- "Index" on page 681

"Sleep" Searches for a sleep frame.
You can refine the search condition, see:

- "NM" on page 681
- "CT" on page 681
- "Data" on page 681
- "Index" on page 681

"Long" Searches for a long frame.
You can refine the search condition, see:

- "Frame ID" on page 680
- "NM" on page 681
- "CT" on page 681
- "DLC Extend" on page 681
- "Data" on page 681
- "Index" on page 681

"Long Poll" Searches for a long poll frame.
You can refine the search condition, see:

- "Frame ID" on page 680
- "NM" on page 681
- "CT" on page 681
- "DLC Extend" on page 681
- "Data" on page 681
- "Index" on page 681

"PID" Searches for a PID field. You can refine the frame identifier, see "Frame ID" on page 680.

"PTYPE" Searches for a PTYPE field

"PTYPE + PID" Searches for a PTYPE field followed by a PID field. You can refine the frame identifier, see "Frame ID" on page 680.

Remote command:
`SEARch:TRIGger:CXPI:TYPE` on page 1952

Frame ID
Searches for a frame identifier pattern or a frame identifier range. The setup consists of the condition and one or two frame patterns.

The frame identifier setup settings are the same as in the CXPI trigger setup, see "Frame ID" on page 676.

Remote command:
`SEARch:TRIGger:CXPI:FID:CONDition` on page 1951
`SEARch:TRIGger:CXPI:FID:MAX` on page 1952
`SEARch:TRIGger:CXPI:FID:MIN` on page 1952

DLC
Searches for a DLC pattern or a DLC range. The setup consists of the condition and one or two DLC patterns.

The DLC setup settings are the same as in the CXPI trigger setup, see "DLC" on page 677.

Remote command:
SEARch:TRIGger:CXPI:DLC:CONDition on page 1949
SEARch:TRIGger:CXPI:DLC:MAX on page 1949
SEARch:TRIGger:CXPI:DLC:MIN on page 1949

DLC Extend
Searches for a DLC extend pattern or a DLC extend range. The setup consists of the condition and one or two DLC extend patterns.

The DLC extend setup settings are the same as in the CXPI trigger setup, see "DLC Extend" on page 677.

Remote command:
SEARch:TRIGger:CXPI:DEXTension:CONDition on page 1948
SEARch:TRIGger:CXPI:DEXTension:MAX on page 1948
SEARch:TRIGger:CXPI:DEXTension:MIN on page 1948

Data
Searches for a data pattern or a data range. The setup consists of the condition and one or two data patterns.

The data setup settings are the same as in the CXPI trigger setup, see "Data" on page 677.

Remote command:
SEARch:TRIGger:CXPI:DATA:DCONdition on page 1946
SEARch:TRIGger:CXPI:DATA:DMAX on page 1946
SEARch:TRIGger:CXPI:DATA:DMIN on page 1946

Index
Searches for a data index pattern or a data index range. The setup consists of the condition and one or two index patterns.

The index setup settings are the same as in the CXPI trigger setup, see "Index" on page 678.

Remote command:
SEARch:TRIGger:CXPI:DATA:ICONdition on page 1947
SEARch:TRIGger:CXPI:DATA:IMAX on page 1947
SEARch:TRIGger:CXPI:DATA:IMIN on page 1947

NM
Searches for the value of the network management (NM), 2 bits indicating if the frame is in wakeup or sleep mode.

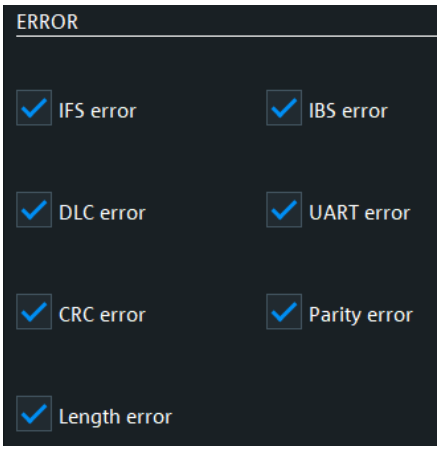
Remote command:
TRIGger<m>:CXPI:NM on page 1940

CT
Searches for the value of the counter (CT), 2 bits indicating the continuity of the frame.

Remote command:
SEARch:TRIGger:CXPI:CT on page 1945

Error
Selects the error type to be searched for. You can select one or more error types as search condition.

The error types are the same as in the CXPI trigger setup, see "Error" on page 675.



13.11.4.2 Search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

The columns in the search result table are the same as in the decoding table, see Chapter 13.11.5, "CXPI decode results", on page 683.

Remote commands:

- `SEARch:RESult:CXPI:FCOunt?` on page 1954
- `SEARch:RESult:CXPI:FRAMe<m>:STATe?` on page 1955
- `SEARch:RESult:CXPI:FRAMe<m>:STARt?` on page 1955
- `SEARch:RESult:CXPI:FRAMe<m>:STOP?` on page 1955
- `SEARch:RESult:CXPI:FRAMe<m>:TYPE?` on page 1955
- `SEARch:RESult:CXPI:FRAMe<m>:DATA?` on page 1954
- `SEARch:RESult:CXPI:FRAMe<m>:DLCV?` on page 1954
- `SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:TYPE?` on page 1956
- `SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:VALue?` on page 1956
- `SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:STATus?` on page 1956

13.11.4.3 Searching CXPI data

Prerequisites: A CXPI bus is configured, see Chapter 13.11.2, "CXPI configuration", on page 667, and "Decode" is enabled.

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Tap "Setup search".
3. Tap the "Source" button and select the serial bus that is set to CXPI.

   "Protocol" shows the CXPI icon.
4. Tap "Type" and select the search type.
   All trigger types are also available for search.
5. To refine the search settings, configure additional settings, which are available for many search types.
   For details, see Chapter 13.11.4.1, "Search settings", on page 679.
6. Under "Control", "Enable" the search.

   The "Search Results" box opens.
7. Close the "Search" dialog box.
8. Press [Run Stop] to start acquisition.
9. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

   The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.11.5 CXPI decode results

When the configuration of the serial bus is complete, the signal can be decoded:

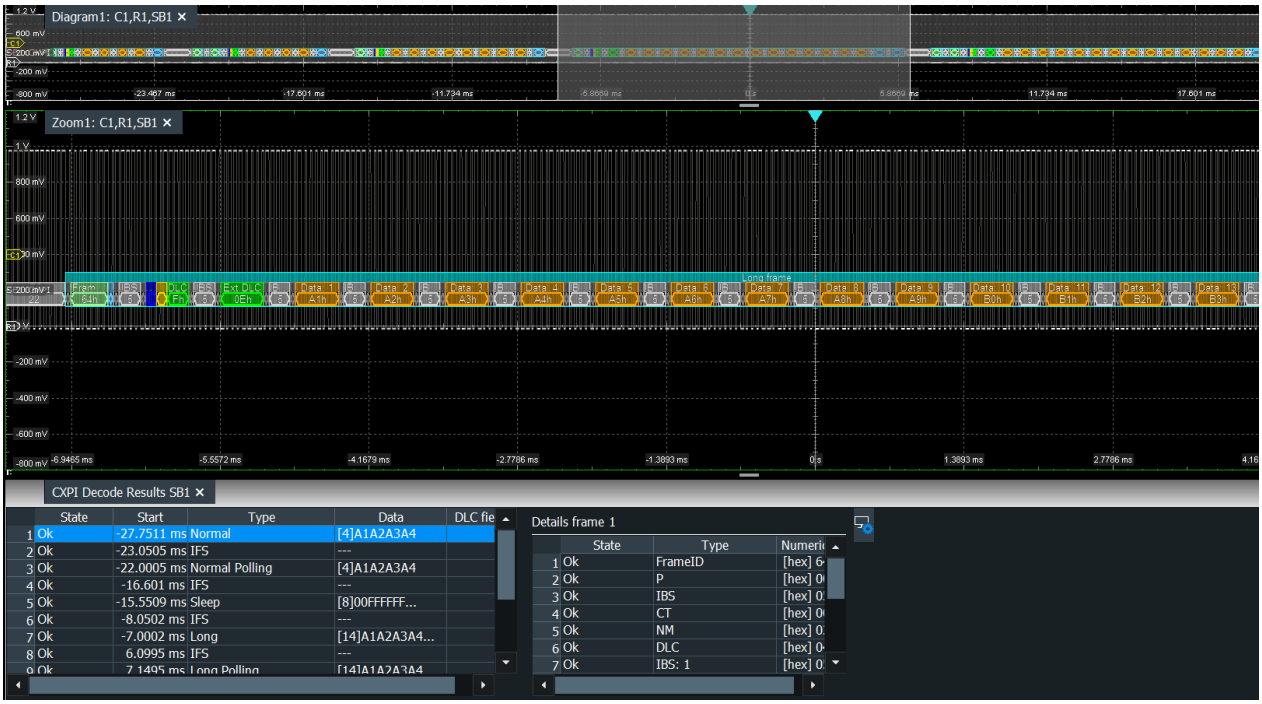
1. In the "Serial Bus" dialog > "Setup" tab, enable "Decode".
2. In the "Serial Bus" dialog > "Display" tab, select additional result display settings: "Show table" and "Show binary".
3. If required, enable "Zoom coupling"

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table



The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

Table 13-16: Content of the Decode results table

Column	Description
State	Overall state of the frame
Start	Time of frame start
Type	Frame type: normal, normal polling, sleep, long, long polling, PID, PTYPE, PTYPE +PID, unknown, inter-frame-space
Data	Displays the first four data words in the frame. More details are displayed in the "Details frame".
DLC field	Data length code, number of data bytes

Table 13-17: Content of the Details frame table

Column	Description
Status	Status of the field: OK or error
Type	Type (name) of field
Numeric value	The numeric value of the field

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands are described in Chapter 24.17.12.3, "Decode results", on page 1941.

13.12 MIL-1553 (option R&S RTO6-K530)

13.12.1 The MIL-STD-1553

The MIL-STD-1553 specification defines the characteristics of a serial data bus originally designed for use in the military avionics. Nowadays it is also used in spacecraft on-board data handling.

The bus is a 2-wire bus that uses differential signals.

A MIL-STD-1553 system consists of the following components:

- Bus Controller (BC): initiates and coordinates the data flow in the system.
- Remote Terminal (RT): interfaces various subsystems with the data bus. A system consists of up to 31 RTs and each RT can have 31 subaddresses.
  The subaddresses 0 and 31 refer to a mode code command.
- Bus Monitor (BM) (optional): listens to all messages and can record selected data for real-time or off-line analysis.

The information is transmitted over the bus in defined series of words using Manchester code, where each bit is transmitted as high-low for a logical 1 or a low-high for a logical 0. There are three types of words: command, data and status.

Command Word

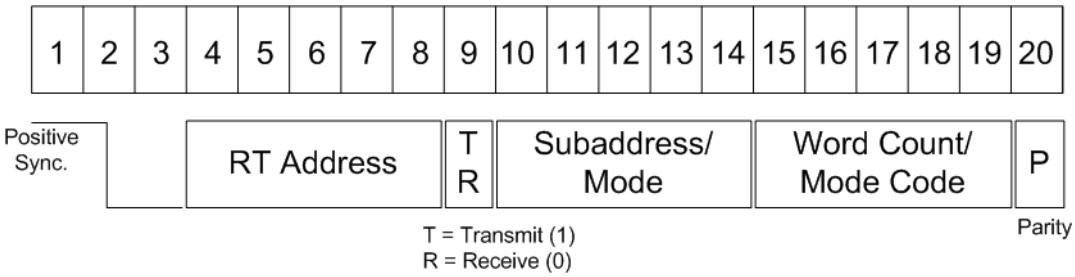


Figure 13-22: Structure of a command word

The format of a command word consists of the following parts (see Figure 13-22):

- Sync: an invalid Manchester waveform.
- Remote Terminal (RT) Address: the unique address of the corresponding RT.
- Transmit/Receive (T/R): indicates the action required from the RT.
- Subaddress/Mode Code: indicates the RT subaddress. The subaddresses 0 and 31 signalize the transmission of a mode code.
- Data Word Count /Mode Code: indicates the number of words that are sent/ received by the RT. A maximum of 32 words is allowed. This field may be used for the transmission of the mode code value.
- Parity: checks if there are bit errors during the transmission. The total number of logic 1 bits for the word (sync bits not included) shall be odd.

Data Word

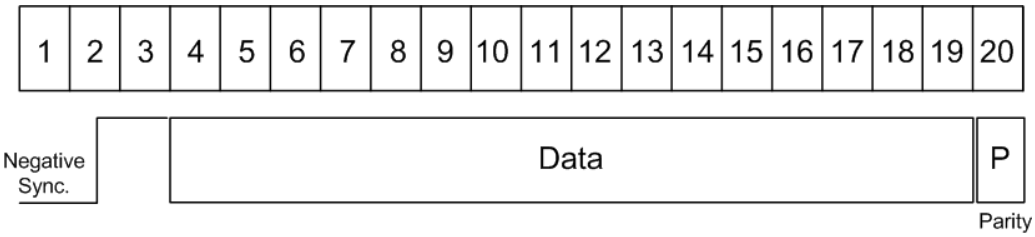


Figure 13-23: Structure of a data word

The format of a data word consists of the following parts (see Figure 13-23):

- Sync: an invalid Manchester waveform.
- Data: the transferred information (16 bit).
- Parity: checks if there are bit errors during the transmission. The total number of logic 1 bits for the word (sync bits not included) shall be odd.

Status Word

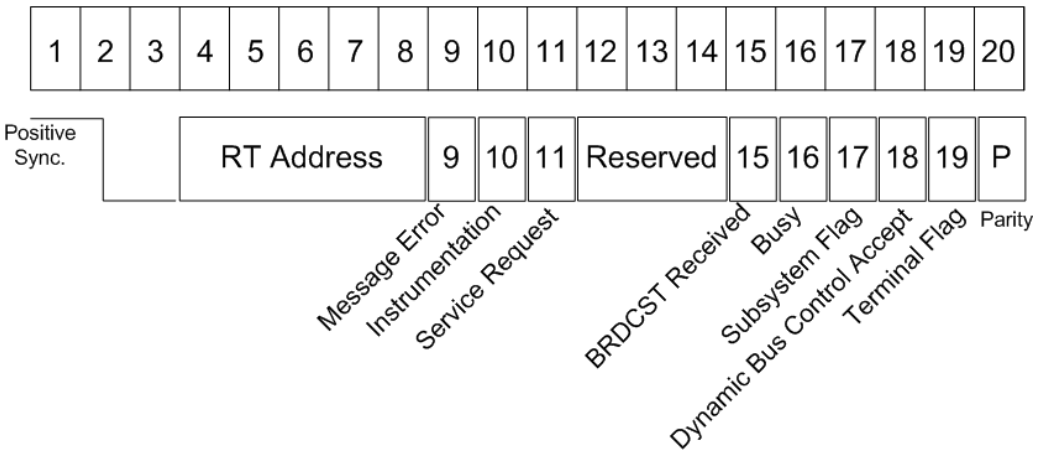


Figure 13-24: Structure of a status word

The format of a status word consists of the following parts (see Figure 13-24):

- Sync: an invalid Manchester waveform.
- Remote Terminal (RT) Address: the unique address of the corresponding RT.
- Message error: indicates an error in the command/data word transmission from the BC. A logic 1 indicates presence of a message error and a logic 0 indicates its absence.
- Instrumentation: helps to distinguish between a status word and a command word. The logic state of this bit shall be 0.
- Service Request: indicates that the RT requires service. A logic 1 indicates a presence of a service request and logic 0 indicates its absence.
- Reserved: bits reserved for future uses.
- Broadcast Command: a logic 1 indicates that the preceding valid command word was a broadcast command and a logic 0 that it was not.
- Busy: a busy state indicates that the RT or the subsystem is not able to transfer data. A logic 1 indicates a presence of a busy condition and logic 0 indicates its absence.
- Subsystem Flag: flags a subsystem fault. A logic 1 indicates a presence of a flag and logic 0 indicates its absence.
- Dynamic Bus Control Acceptance: a logic 1 indicates acceptance of a dynamic bus control and a logic 0 a rejection.
- Terminal Flag: flags an RT fault condition. A logic 1 indicates a presence of a flag and logic 0 indicates its absence.
- Parity: checks if there are bit errors during the transmission. The total number of logic 1 bits for the word (sync bits not included) shall be odd.

For comfortable analysis, you can load an editable label list, to interpret transferred numeric values as meaningful text labels.

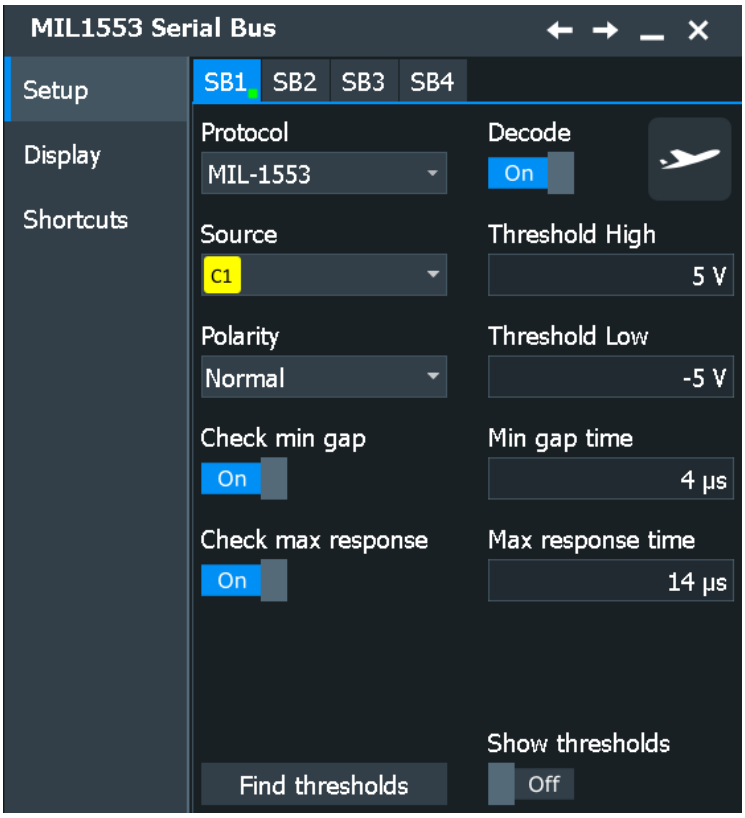
13.12.2 MIL-STD-1553 configuration

13.12.2.1 MIL-STD-1553 configuration settings

Access: [Bus] > "Decode" tab > "Protocol" = *MIL-STD-1553*

Access: [Bus] > "MIL-1553" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Source
Sets the source of the selected data line. Waveform 1 of channel signals, math waveforms, and reference waveforms can be used.

For triggering on a serial bus, a channel signal is required.

Remote command:
`BUS<m>:MILStd:SOURce` on page 1957

Polarity
Selects the wire on which the bus signal is measured : "Normal" or "Inverted". The setting affects the digitization of the signal.

Remote command:
`BUS<m>:MILStd:POLarity` on page 1959

Thresholds
Threshold values are used for digitization of the signal.

Sets the threshold value for digitization of signals for each line. If the signal value on the line is higher than the threshold, the signal state is high (1 or true for the Boolean logic). Otherwise, the signal state is considered low (0 or false) if the signal value is below the threshold.

"Threshold high" and "Threshold low"	Upper and lower threshold levels. You can enter the values directly in the fields.
"Find thresholds"	Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
`BUS<m>:MILStd:THReshold:HIGH` on page 1959
`BUS<m>:MILStd:THReshold:LOW` on page 1959
`BUS<m>:MILStd:PRESet` on page 1959
`BUS<m>:SETReflevels` on page 1671

Check min gap
Checks and sets a value for the intermessage gap between the last bit of a message and the following command word sync. The time is measured between the mid bit zero crossings. According to the standard, the minimum idle time is 4 µs.

The minimum gap time is relevant for protocol configuration and error trigger.

If "Check min gap" is enabled, the instrument detects the specified gap during decoding. If the trigger type "Error condition" is selected in addition, the instrument triggers when the gap is shorter then specified.

Remote command:
`BUS<m>:MILStd:MINGap:BITS` on page 1958
`BUS<m>:MILStd:MINGap:SELect` on page 1958
`TRIGger<m>:MILStd:MINGap:BITS` on page 1968
`TRIGger<m>:MILStd:MINGap:SELect` on page 1968

Check max response
Checks and sets a value for the maximum response time between the last bit of a word and the following status word sync. The time is measured between the mid bit zero crossings. According to the standard, the RT shall respond to a valid command word within the time period of 4 to 12 µs.

The max response time is relevant for protocol configuration and error trigger.

If "Check max response" is enabled, the instrument detects the specified gap during decoding. If the trigger type "Error condition" is selected in addition, the instrument triggers when the response time is longer than specified.

Remote command:
BUS<m>:MILStd:MAXResponse:BITS on page 1957
BUS<m>:MILStd:MAXResponse:SELect on page 1958
TRIGger<m>:MILStd:MAXResponse:BITS on page 1967
TRIGger<m>:MILStd:MAXResponse:SELect on page 1967

Show thresholds
If enabled, the threshold levels are displayed in the diagram.

Remote command:
BUS<m>:THReshold on page 1671

13.12.2.2 Display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

MIL Specific Label Settings

Label lists are protocol-specific. A MIL-STD-1553 label file contains four values for each identifier:

- "RTA": hexadecimal remote terminal address value
- "Sub Addr": hexadecimal sub address value
- "Sub Address Label Name": the label name corresponding to the value of the sub-address.
- "Symbolic label": symbolic name of addressed device, specifying the device function, and the label of the sub address.

Example: MIL PTT file

```
# ------------------------------------------------------------
# Labels for MIL.1553 protocol
# Column order: RT address, RT label, Subaddress, Subaddress Label
# ---------------------------------------------------------------------
@PROTOCOL_NAME = mil1553
0Ah,Engine,01h,Thrust
03h,Main panel,07h,Altimeter
03h,Main panel,01h,Speed
0Eh,Only RTA
```

13.12.2.3 Configuring MIL-STD-1553

For configuration assign the line to the input channel, set the threshold and the timing conditions.

For details on configuration settings, see Chapter 13.12.2.1, "MIL-STD-1553 configuration settings", on page 688.

1. Press the [Bus] key on the front panel.

2. Select the tab of the bus you want to set up, for example "SB1".
3. In the "Setup" tab, select the protocol: "MIL-STD-1553".
4. Tap the "Polarity" button, and select the waveform of the data line.
5. Set the logical thresholds.
6. If necessary, tap the "Check min gap" button to select it and set the minimum gap time.
7. If necessary, tap the "Check max response" button to select it and set the maximum response time.

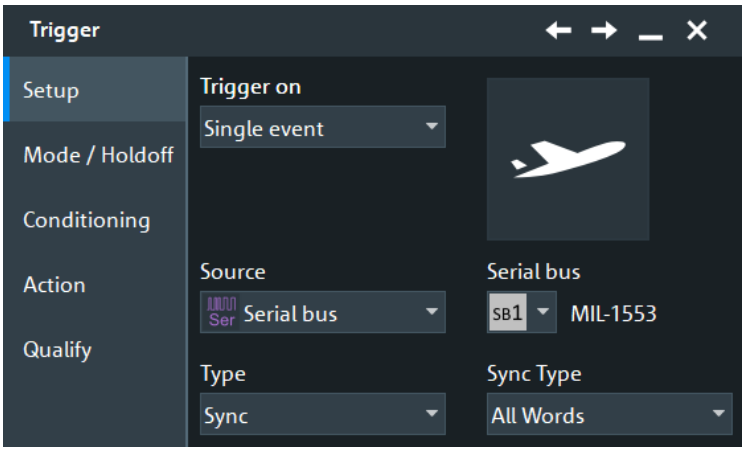
13.12.3 MIL-STD-1553 trigger

13.12.3.1 Trigger settings MIL-STD-1553

Access: [Bus] > "Shortcuts" tab > "Setup trigger"

Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

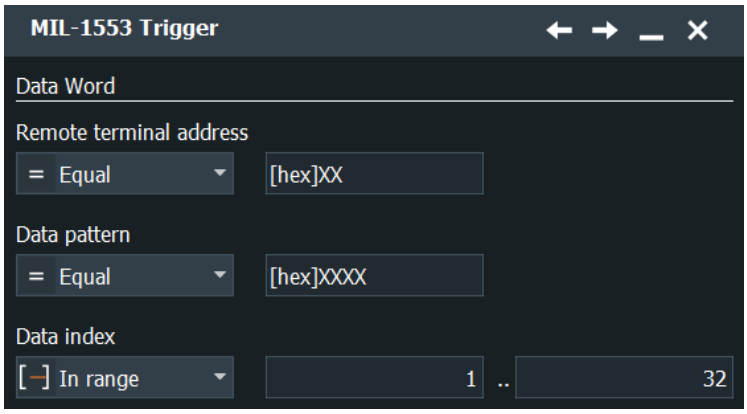


Trigger Type
Selects the trigger type for MIL-STD-1553 analysis.

"Sync" Triggers on a sync impulse.

"Word" Triggers on the selected word type.

"Data Word" Triggers on a specified data word or data word range.

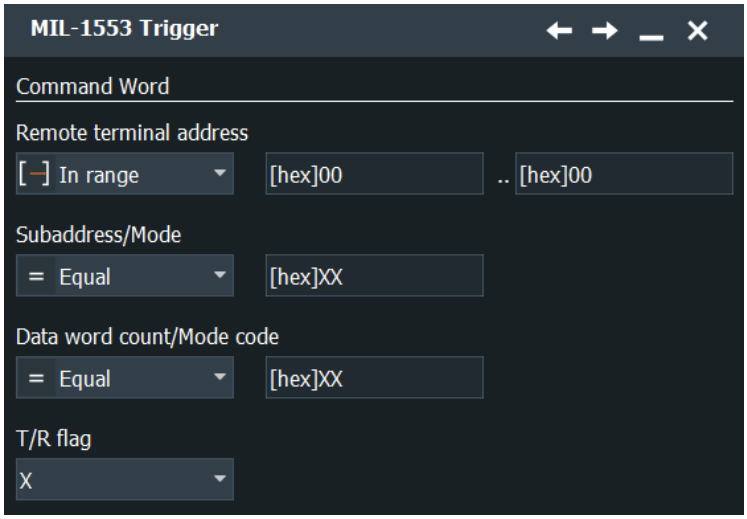


"Command/ Status Word"

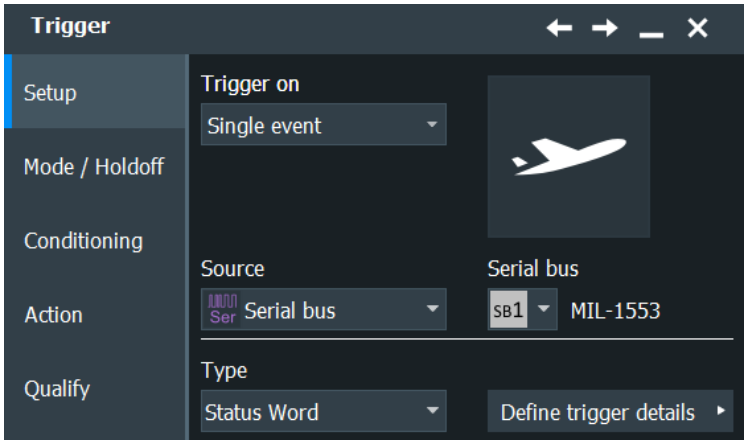
Triggers on a specified command word or on a status word.

"Command Word"

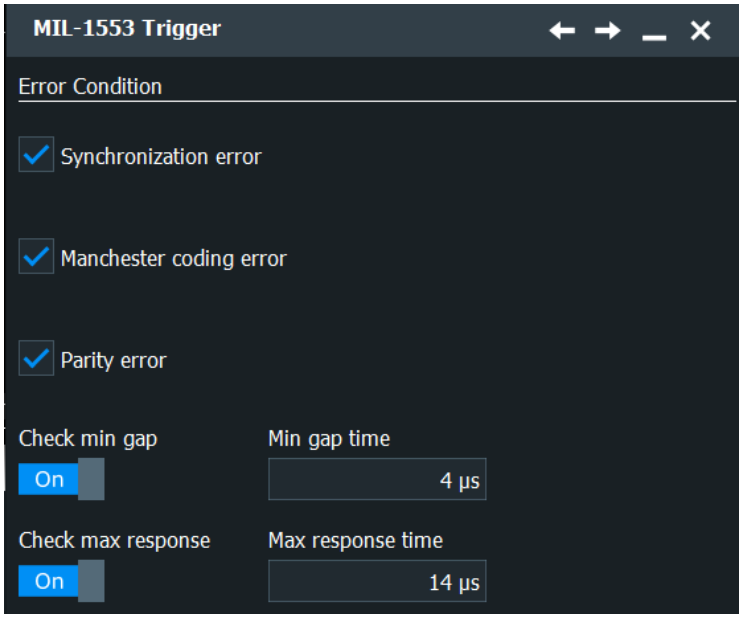
Triggers on a specified command word.



"Status Word" Triggers on a specified status word.



"Error Condition" Triggers on any combination of protocol errors.



Remote command:
`TRIGger<m>:MILStd:TYPE` on page 1961

Sync Type / Word Type
Triggers on a sync impulse or word type. You can select to trigger on "Command/ Status", on "All" or on "Data" sync pulses / word types.

Remote command:
`TRIGger<m>:MILStd:TPSPecifier` on page 1969

Remote Terminal Address
The RTA setup consists of the condition and one or two RTA patterns.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Condition" Sets the comparison condition to a specific value or a range.

"RTA Min/ RTA" Specifies the value or sets the start value of a range.

"RTA Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:MILStd:CDST:RCONdition` on page 1961
`TRIGger<m>:MILStd:CDST:RMAX` on page 1962
`TRIGger<m>:MILStd:CDST:RMIN` on page 1962
`TRIGger<m>:MILStd:CMD:RCONdition` on page 1961
`TRIGger<m>:MILStd:CMD:RMAX` on page 1962
`TRIGger<m>:MILStd:CMD:RMIN` on page 1962
`TRIGger<m>:MILStd:DATA:RCONdition` on page 1961
`TRIGger<m>:MILStd:DATA:RMAX` on page 1962
`TRIGger<m>:MILStd:DATA:RMIN` on page 1962

Data Pattern
Sets the data to be triggered on. The trigger data setup consists of a comparison condition and one or two data patterns.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Condition" Sets the comparison condition to a specific value or a range.

"Data Min/ Data" Specifies the value or sets the start value of a range.

"Data Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:MILStd:DATA:DCONdition` on page 1965
`TRIGger<m>:MILStd:DATA:DMAX` on page 1965
`TRIGger<m>:MILStd:DATA:DMIN` on page 1965

Data Index
The MIL-MIL-STD-1553 standard defines the length of a message to a series of up to 32 words. Data index sets the range within this series of the data words that is considered for the analysis. The data index setup consists of the condition and one or two data index values.

"Condition" Sets the comparison condition to a specific value or a range.

"Index Min/Index" Specifies the value or sets the start value of a range.

"Index Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:MILStd:DATA:ICONdition` on page 1966
`TRIGger<m>:MILStd:DATA:IMAX` on page 1966
`TRIGger<m>:MILStd:DATA:IMIN` on page 1966

11-Bit Information
The 11-Bit information sets bits 9 to 19 if there is a command or status word. The 11-Bit information consists of the condition and one or two 11-Bit information patterns.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Condition" Sets the comparison condition to a specific value or a range.

"Info Min/Info" Specifies the value or sets the start value of a range.

"Info Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:MILStd:CDST:ICONdition` on page 1964
`TRIGger<m>:MILStd:CDST:IMAX` on page 1965
`TRIGger<m>:MILStd:CDST:IMIN` on page 1965

Subaddress/ Mode
The subaddress/mode setup consists of the condition and one or two subaddress/ mode patterns.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Condition" Sets the comparison condition to a specific value or a range.

"Subaddress Min / Subaddress/Mode"
Specifies the value or sets the start value of a range.

"Subaddress Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:MILStd:CMD:SCONdition` on page 1963
`TRIGger<m>:MILStd:CMD:SMAX` on page 1963
`TRIGger<m>:MILStd:CMD:SMIN` on page 1964

Data Word Count/Mode Code
The data word count/ mode code setup consists of the condition and one or two patterns.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Condition" Sets the comparison condition to a specific value or a range.

"Word Count Min/ Count Code"
Specifies the value or sets the start value of a range.

"Word Count Max"
Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:MILStd:CMD:CCONdition` on page 1962
`TRIGger<m>:MILStd:CMD:CMAX` on page 1963
`TRIGger<m>:MILStd:CMD:CMIN` on page 1963

T/R (Transmit/receive)
Toggles the data direction of the selected command: 1 (transmit), 0 (receive), or X (either).

Remote command:
`TRIGger<m>:MILStd:CMD:TR` on page 1964

Status Flags
Specifies the values of the status flags. You can use the following characters: 1; 0; or X (do not care).

For details, see "Status Word" on page 687.

Remote command:
`TRIGger<m>:MILStd:STATus:BCReceived` on page 1968
`TRIGger<m>:MILStd:STATus:BUSY` on page 1968
`TRIGger<m>:MILStd:STATus:DBCaccept` on page 1968
`TRIGger<m>:MILStd:STATus:INSTrument` on page 1969
`TRIGger<m>:MILStd:STATus:MERRor` on page 1969
`TRIGger<m>:MILStd:STATus:SREQuest` on page 1969
`TRIGger<m>:MILStd:STATus:SUBSystem` on page 1969
`TRIGger<m>:MILStd:STATus:TERMinal` on page 1969

Error Condition
Specify the error conditions to be triggered on.

Synchronization Error ← Error Condition
Triggers if a sync impulse does not fulfill the technical requirements or when the transmission is not valid.

Remote command:
`TRIGger<m>:MILStd:ERRor:SYNC` on page 1967

Manchester Coding Error ← Error Condition
Triggers if there is an error in the Manchester coding of the signal.

Remote command:
`TRIGger<m>:MILStd:ERRor:MANChester` on page 1966

Parity Error ← Error Condition
Checks the parity of every word and triggers if the parity is even.

Remote command:
`TRIGger<m>:MILStd:ERRor:PARity` on page 1967

Check min gap ← Error Condition
Checks and sets a value for the intermessage gap between the last bit of a message and the following command word sync. The time is measured between the mid bit zero crossings. According to the standard, the minimum idle time is 4 µs.

The minimum gap time is relevant for protocol configuration and error trigger.

If "Check min gap" is enabled, the instrument detects the specified gap during decoding. If the trigger type "Error condition" is selected in addition, the instrument triggers when the gap is shorter then specified.

Remote command:
`BUS<m>:MILStd:MINGap:BITS` on page 1958
`BUS<m>:MILStd:MINGap:SELect` on page 1958
`TRIGger<m>:MILStd:MINGap:BITS` on page 1968
`TRIGger<m>:MILStd:MINGap:SELect` on page 1968

Check max response ← Error Condition
Checks and sets a value for the maximum response time between the last bit of a word and the following status word sync. The time is measured between the mid bit zero crossings. According to the standard, the RT shall respond to a valid command word within the time period of 4 to 12 µs.

The max response time is relevant for protocol configuration and error trigger.

If "Check max response" is enabled, the instrument detects the specified gap during decoding. If the trigger type "Error condition" is selected in addition, the instrument triggers when the response time is longer than specified.

Remote command:
`BUS<m>:MILStd:MAXResponse:BITS` on page 1957
`BUS<m>:MILStd:MAXResponse:SELect` on page 1958
`TRIGger<m>:MILStd:MAXResponse:BITS` on page 1967
`TRIGger<m>:MILStd:MAXResponse:SELect` on page 1967

13.12.3.2 Triggering on MIL-STD-1553

Prerequisites: A bus is configured for the MIL-STD-1553 signal to be analyzed.

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to MIL-STD-1553.
5. Select the "Trigger type".
6. For more complex trigger types, enter the data pattern conditions.
   For details, see Chapter 13.12.3.1, "Trigger settings MIL-STD-1553", on page 691.

13.12.4 MIL-STD-1553 decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

Table 13-18: Content of the decode result table

Column	Description
State	Overall state of the word
Start	Time of word start in relation to the trigger point
Stop	Time of word stop in relation to the trigger point
Type	Word type
RTA	Remote terminal address
Info	The hexadecimal value of the 9th to 1th bit of a command/status word
Label	Symbolic label name defined in the label list
Data	The values of the data bytes

Remote commands

Remote commands to retrieve decode results are described in Chapter 24.17.13.3, "Decode results", on page 1970.

13.12.5 Search on decoded MIL data

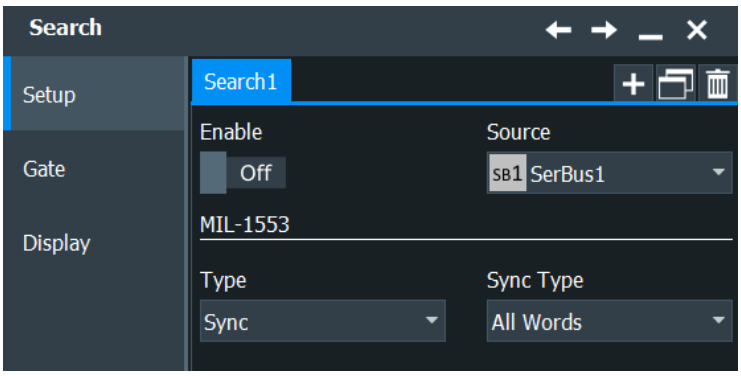
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 11, "Search functions", on page 419.

13.12.5.1 MIL search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"



Type
The search criterion is defined by "Type". All trigger types are also available for search.

Additional search parameters are provided under "Detailed Search Parameter Setup".

"Sync"	Searches for a sync impulse.
"Word"	Searches for the selected word type.
"Data Word"	Searches for the specified data word. Additional search parameters: remote terminal address, data pattern and data index.
"Command / Status Word"	Searches for command or status words. Additional search parameters: remote terminal address, and 11-bit information.
"Command Word"	Searches for a command word. Additional search parameters: remote terminal address, subaddress / mode, and data word count / mode code.
"Status Word"	Searches for a status word. Additional search parameters: remote terminal address, and status flags.
"Error condition"	Identifies various errors in the frame, see "Error Condition" on page 696.

Remote command:
`SEARch:TRIGger:MILStd:TYPE` on page 1974

Sync Type / Word Type
Searches for a sync impulse/ word type. You can search for "Command/Status", "All" or "Data" sync pulses/ word types.

Remote command:
`SEARch:TRIGger:MILStd:TPSPecifier` on page 1979

Remote terminal address setup: Condition, RTA
The remote terminal address setup consists of the condition and one or two RTA patterns.

The RTA setup settings are the same as in the MIL trigger setup, see "Remote Terminal Address" on page 693.

Remote command:
`SEARch:TRIGger:MILStd:CDST:RCONdition` on page 1974
`SEARch:TRIGger:MILStd:CMD:RCONdition` on page 1974
`SEARch:TRIGger:MILStd:DATA:RCONdition` on page 1974
`SEARch:TRIGger:MILStd:CDST:RMIN` on page 1975
`SEARch:TRIGger:MILStd:CMD:RMIN` on page 1975
`SEARch:TRIGger:MILStd:DATA:RMIN` on page 1975
`SEARch:TRIGger:MILStd:CDST:RMAX` on page 1975
`SEARch:TRIGger:MILStd:CMD:RMAX` on page 1975
`SEARch:TRIGger:MILStd:DATA:RMAX` on page 1975

Data pattern setup: Condition, Data min, Data max
The data pattern setup consists of the condition and one or two data patterns.

The data pattern setup settings are the same as in the MIL trigger setup, see "Data Pattern" on page 694.

Remote command:
`SEARch:TRIGger:MILStd:DATA:DCONdition` on page 1976
`SEARch:TRIGger:MILStd:DATA:DMIN` on page 1976
`SEARch:TRIGger:MILStd:DATA:DMAX` on page 1977

Data index setup: Condition, Index min, Index max
The data index setup consists of the condition and one or two index patterns.

The data index setup settings are the same as in the MIL trigger setup, see "Data Index" on page 694.

Remote command:
`SEARch:TRIGger:MILStd:DATA:ICONdition` on page 1977
`SEARch:TRIGger:MILStd:DATA:IMIN` on page 1977
`SEARch:TRIGger:MILStd:DATA:IMAX` on page 1978

11-Bit information setup: Condition, Info min, Info max
The 11-bit information setup consists of the condition and one or two 11-bit information patterns.

The 11-bit information setup settings are the same as in the MIL trigger setup, see "11-Bit Information" on page 695.

Remote command:
`SEARch:TRIGger:MILStd:CDST:ICONdition` on page 1976
`SEARch:TRIGger:MILStd:CDST:IMIN` on page 1976
`SEARch:TRIGger:MILStd:CDST:IMAX` on page 1977

Subaddress / Mode setup: Condition, Subaddress min, Subaddress max
The subaddress/mode setup consists of the condition and one or two subaddress/mode patterns.

The subaddress/mode setup settings are the same as in the MIL trigger setup, see "Subaddress/ Mode" on page 695.

Remote command:
`SEARch:TRIGger:MILStd:CMD:SCONdition` on page 1976
`SEARch:TRIGger:MILStd:CMD:SMIN` on page 1976
`SEARch:TRIGger:MILStd:CMD:SMAX` on page 1977

Data word count / Mode code setup: Condition, Word count min, Word count max
The data word count/mode code setup consists of the condition and one or two patterns.

The subaddress/mode setup settings are the same as in the MIL trigger setup, see "Data Word Count/Mode Code" on page 695.

Remote command:
`SEARch:TRIGger:MILStd:CMD:CCONdition` on page 1976
`SEARch:TRIGger:MILStd:CMD:CMIN` on page 1976
`SEARch:TRIGger:MILStd:CMD:CMAX` on page 1977

T/R (Transmit/receive)
Specifies the data direction of the selected command.

For details, see "T/R (Transmit/receive)" on page 696.

Remote command:
`SEARch:TRIGger:MILStd:CMD:TR` on page 1978

Status flags setup
Specifies the values (X, 0, 1) of the status flags.

The status flags setup settings are the same as in the MIL trigger setup, see "Status Flags" on page 696.

Remote command:
`SEARch:TRIGger:MILStd:STATus:BCReceived` on page 1979
`SEARch:TRIGger:MILStd:STATus:BUSY` on page 1979
`SEARch:TRIGger:MILStd:STATus:DBCaccept` on page 1979
`SEARch:TRIGger:MILStd:STATus:INSTrument` on page 1979
`SEARch:TRIGger:MILStd:STATus:MERRor` on page 1979
`SEARch:TRIGger:MILStd:STATus:SREQuest` on page 1979
`SEARch:TRIGger:MILStd:STATus:SUBSystem` on page 1979
`SEARch:TRIGger:MILStd:STATus:TERMinal` on page 1979

Error Condition
Selects the error type to be searched for. You can select one or more error types as search condition.

The error types are the same as in the MIL trigger setup, see "Error Condition" on page 696

Remote command:
`SEARch:TRIGger:MILStd:ERRor:MANChester` on page 1978
`SEARch:TRIGger:MILStd:ERRor:PARity` on page 1978
`SEARch:TRIGger:MILStd:ERRor:SYNC` on page 1978
`SEARch:TRIGger:MILStd:ERRor:TIMing` on page 1978

13.12.5.2 MIL search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

The serach results are the same as Chapter 13.12.4, "MIL-STD-1553 decode results", on page 698

Remote commands:

- `SEARch:RESult:MILStd:WCOunt?` on page 1980
- `SEARch:RESult:MILStd:WORD<m>:INFO?` on page 1982
- `SEARch:RESult:MILStd:WORD<m>:RTADdress?` on page 1982
- `SEARch:RESult:MILStd:WORD<m>:STARt?` on page 1981
- `SEARch:RESult:MILStd:WORD<m>:STATus?` on page 1980
- `SEARch:RESult:MILStd:WORD<m>:STOP?` on page 1981
- `SEARch:RESult:MILStd:WORD<m>:SYMBol?` on page 1981
- `SEARch:RESult:MILStd:WORD<m>:TYPE?` on page 1980

13.13 ARINC 429 (option R&S RTO6-K530)

13.13.1 ARINC 429 basics

The ARINC 429 is a specification that defines the characteristics of an avionic data bus used on commercial and transport aircraft.

In an ARINC 429 system, a single transmitter/source is connected to 1 to 20 receivers/sinks on one twisted wire pair. The bus uses differential signals. The ARINC 429 standard uses a simplex communication - data may be transmitted in only one direction. The information is transmitted over the bus in defined series of words.

Word Format

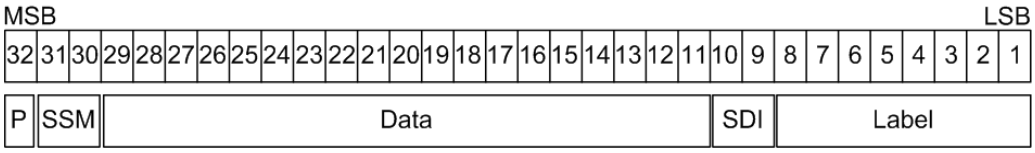


Figure 13-25: Structure of an ARINC 429 word

An ARINC 429 word is 32-bits and consists of the following parts (see Figure 13-25):

- Parity: the most significant bit (MSB). Checks if there are bit errors during the transmission. The total number of logic 1 bits for the word shall be odd.
- Sign/Status Matrix (SSM): the value of these bits depends on the data type. It may be used to report the status of hardware equipment.
- Data:
  - Binary (BNR): stores the data as a binary number.
  - Binary Coded Decimal (BCD): uses 4 data field bits to represent a decimal digit.
  - Discrete data: a combination of BNR and/ or BCD or individual bits that express specific equipment conditions.
  - Maintenance data and acknowledgment
  - Williamsburg / Buckhorn protocol: a bit-oriented protocol that is used for file transfer.
- Source/Destination Identifier (SDI): indicates the intended receiver or the transmitting subsystem.
- Label: gives information about the word's data type.

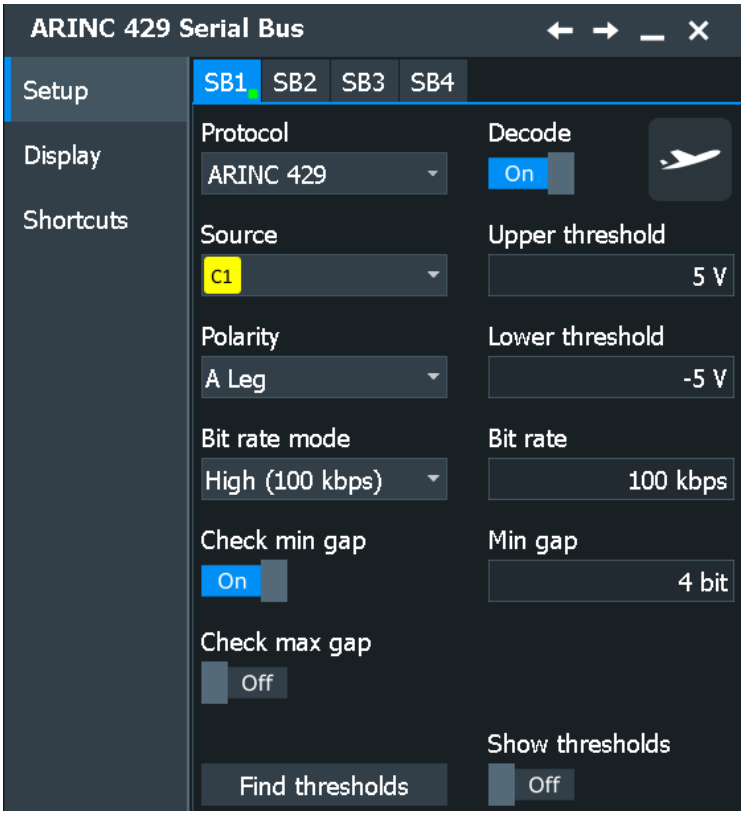
For comfortable analysis, you can load an editable label list, to interpret transferred numeric values as meaningful text labels.

13.13.2 ARINC 429 configuration

13.13.2.1 ARINC 429 configuration settings

Access: [Bus] > "ARINC 429" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Data
Sets the source of the selected data line. Usually, the source is one of the analog channels. Reference and math waveforms are only available if the trigger source is one of the input channels but not the serial bus.

For triggering on a serial bus, a channel signal is required.

Remote command:
`BUS<m>:ARINc:SOURce` on page 1983

Polarity
Selects the wire on which the bus signal is measured : "A Leg" or "B Leg". The setting affects the digitization of the signal.

Remote command:
`BUS<m>:ARINc:POLarity` on page 1985

Thresholds
Sets the threshold value for digitization of the data signal. If the signal value on the line is higher than the threshold, the signal state is high (1 or true for the Boolean logic). Otherwise, the signal state is considered low (0 or false) if the signal value is below the threshold.

There are several ways to set the thresholds:
- "Upper threshold" and "Lower threshold"
  Upper and lower threshold levels. You can enter the values directly in the fields.

- "Find thresholds"

Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
`BUS<m>:ARINc:THReshold:HIGH` on page 1985
`BUS<m>:ARINc:THReshold:LOW` on page 1985
`BUS<m>:ARINc:PRESet` on page 1985
`BUS<m>:SETReflevels` on page 1671

Bit Rate
Selects the number of transmitted bits per second. The value can be set to high speed (100 kbps) or low speed (12.0- 14.5 kbps).

Remote command:
`BUS<m>:ARINc:BRValue` on page 1983
`BUS<m>:ARINc:BRMode` on page 1983

Timing: Min gap, Max gap
Defines the idle time between two words, which is needed for word synchronization. The beginning of the first bit after the gap marks the start of a new word.

You can define a minimum idle time "Min gap", and/or a maximum time "Max gap". The standard defines a minimum of 4-bit times to separate two subsequent words.

Timing settings are relevant for protocol configuration and error trigger.

If "Min gap" and/or "Max gap" are enabled, the instrument detects the specified gaps during decoding. If the trigger type "Error condition" is selected in addition, the instrument triggers when the gap is shorter or longer than the specified gaps, respectively.

Remote command:
`BUS<m>:ARINc:MAXGap:BITS` on page 1984
`BUS<m>:ARINc:MAXGap:SELect` on page 1984
`BUS<m>:ARINc:MINGap:BITS` on page 1984
`BUS<m>:ARINc:MINGap:SELect` on page 1984
`TRIGger<m>:ARINc:MINGap:BITS` on page 1988
`TRIGger<m>:ARINc:MINGap:SELect` on page 1988
`TRIGger<m>:ARINc:MAXGap:BITS` on page 1989
`TRIGger<m>:ARINc:MAXGap:SELect` on page 1989

Show thresholds
If enabled, the threshold levels are displayed in the diagram.

Remote command:
`BUS<m>:THReshold` on page 1671

13.13.2.2 Display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

ARINC 429 Specific Label Settings

Label lists are protocol-specific. An ARINC 429 label file contains two values for each identifier:

- "Arinc Label": the ARINC 429 label value, that identifies the data type and the parameters associated with it. The usual data format is octal.
- "Symbolic label": symbolic name of the label, specifying its function.

Example: ARINC 429 PTT file

```
# ------------------------------------------------------------------------------
@FILE_VERSION = 1.0
@PROTOCOL_NAME = arinc429
# ------------------------------------------------------------------------------
# Labels for ARINC 429 protocol
#   Column order: Arinc Label, Symbolic Label
# ------------------------------------------------------------------------------
# ----Definition----
001o, Distance to Go
002o, Time to Go
010o, Present Position - Latitude
011o, Present Position - Longitude
014o, Magnetic Heading
015o, Wind Speed
075o, Gross Weight
125o, Universal Time Coordinated
# ------------------------------------------------------------------------------
s
```

13.13.2.3 Configuring ARINC 429 signals

For configuration assign the line to the input channel, set the threshold and the timing conditions.

For details on configuration settings, see Chapter 13.13.2.1, "ARINC 429 configuration settings", on page 703.

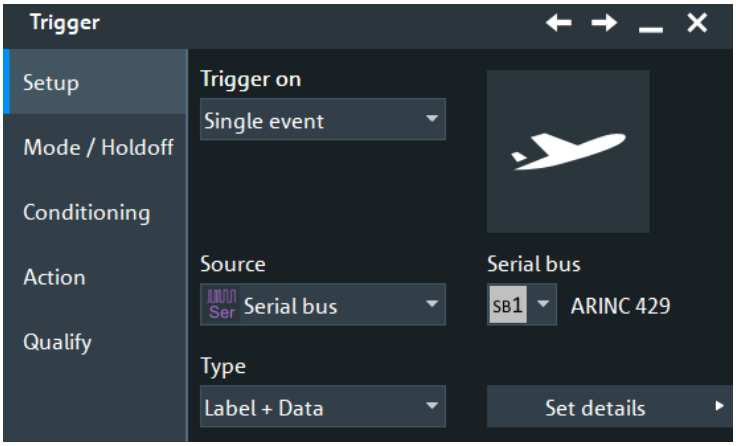
1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. In the "Setup" tab, select the protocol: "ARINC 429".
4. Tap the "Polarity" button, and select the waveform of the data line.
5. Set the logical thresholds.
6. Tap the "Bit Rate" button and set it for high or low speed.
7. If necessary, tap the "Min Gap" button to select it and set the minimum gap time.
8. If necessary, tap the "Max Response" button to select it and set the maximum response time.

9. Enable "Decode".

13.13.3 ARINC 429 trigger

13.13.3.1 ARINC 429 trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



Make sure that:

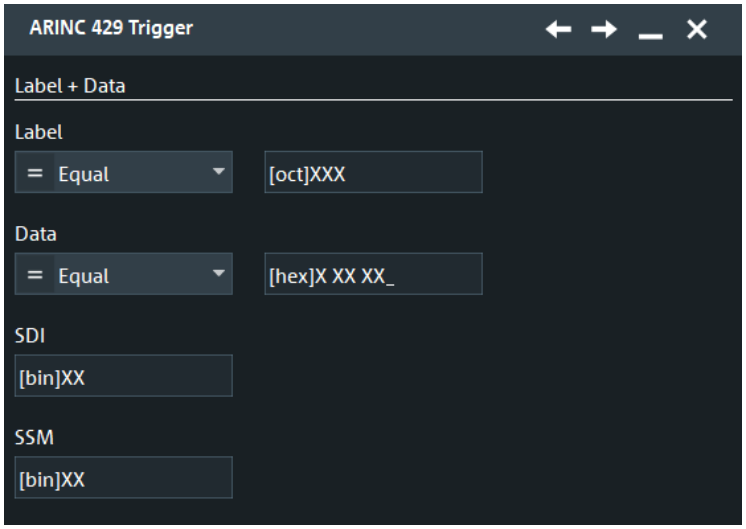
- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Trigger Type
Selects the trigger type for ARINC 429 analysis.

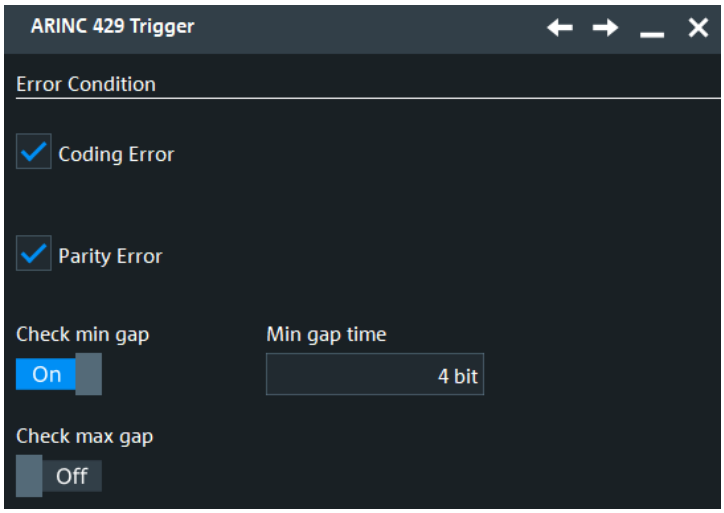
"Word Start" Sets the trigger to the start of the word.

"Word Stop" Sets the trigger to the stop of the word.

"Label + Data" Sets the trigger on a defined word format. You can define the label, the data and the SDI / SSM bits separately, see "Label + Data" on page 708.



"Error Condition"
Identifies various errors in the word, see "Error Conditions" on page 709.



Remote command:
`TRIGger<m>:ARINc:TYPE` on page 1986

Label + Data
Sets the trigger on a defined word format. You can define the label, the data and the SDI / SSM bits separately.

Label setup: Condition, Label Min, Label Max ← Label + Data
The label setup consists of the condition and one or two label patterns.

"Condition" Defines the operator to set a specific label ("Equal" or "Not equal") or a label range.

"Label Min" Defines the bit pattern of the label.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Label Max" The second label pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:ARINc:LABel:CONDition` on page 1988
`TRIGger<m>:ARINc:LABel:MIN` on page 1988
`TRIGger<m>:ARINc:LABel:MAX` on page 1988

Data setup: Condition, Data Min, Data Max ← Label + Data
The data setup consists of the condition and one or two data patterns.

"Condition" Defines the operator to set a specific data ("Equal" or "Not equal") or a data range.

"Data Min" Defines the bit pattern of the data.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:ARINc:DATA:CONDition` on page 1986
`TRIGger<m>:ARINc:DATA:MIN` on page 1987
`TRIGger<m>:ARINc:DATA:MAX` on page 1987

SDI / SSM ← Label + Data
Sets the values for the source/destination identifier (SDI) and the sign/status matrix (SSM) bits.

Remote command:
`TRIGger<m>:ARINc:SDI` on page 1989
`TRIGger<m>:ARINc:SSM` on page 1989

Error Conditions
Specifies the error conditions to be triggered on.

Coding error ← Error Conditions
Triggers on a coding error.

Remote command:
`TRIGger<m>:ARINc:ERRor:CODing` on page 1987

Parity Error ← Error Conditions
Checks the parity and triggers if the parity is even.

Remote command:
`TRIGger<m>:ARINc:ERRor:PARity` on page 1987

Timing: Min gap, Max gap ← Error Conditions
Defines the idle time between two words, which is needed for word synchronization. The beginning of the first bit after the gap marks the start of a new word.

You can define a minimum idle time "Min gap", and/or a maximum time "Max gap". The standard defines a minimum of 4-bit times to separate two subsequent words.

Timing settings are relevant for protocol configuration and error trigger.

If "Min gap" and/or "Max gap" are enabled, the instrument detects the specified gaps during decoding. If the trigger type "Error condition" is selected in addition, the instrument triggers when the gap is shorter or longer than the specified gaps, respectively.

Remote command:
`BUS<m>:ARINc:MAXGap:BITS` on page 1984
`BUS<m>:ARINc:MAXGap:SELect` on page 1984
`BUS<m>:ARINc:MINGap:BITS` on page 1984
`BUS<m>:ARINc:MINGap:SELect` on page 1984
`TRIGger<m>:ARINc:MINGap:BITS` on page 1988
`TRIGger<m>:ARINc:MINGap:SELect` on page 1988
`TRIGger<m>:ARINc:MAXGap:BITS` on page 1989
`TRIGger<m>:ARINc:MAXGap:SELect` on page 1989

13.13.3.2 Triggering on ARINC 429

Prerequisites: A bus is configured for the ARINC 429 signal to be analyzed.

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to ARINC 429.
5. Select the "Trigger type".
6. For more complex trigger types, enter the data pattern conditions.
   For details, see Chapter 13.13.3.1, "ARINC 429 trigger settings", on page 707.

13.13.4 ARINC 429 decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".

2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The "Decode results" table contains information about all decoded frames.

ARINC 429 Decode Results SB1

	State	Start	Stop	Label [oct]	SDI [bin]	Data [hex]	SSM [bin]	Label name	Data (32 bit)
1	Ok	-1.69 ms	-1.37 ms	101	10	2AAAA	11	Bus	[hex] EAAA AA82
2	Coding Error	-1.32 ms	-1.22 ms	000	00	00	00		[hex] 0000 0000
3	Ok	-950 µs	-630 µs	101	01	54455	11	Bus	[hex] 7511 5582
4	Ok	-580 µs	-260 µs	101	01	C955	11	Bus	[hex] 6325 5582
5	Ok	-210 µs	110 µs	101	10	22222	10	Bus	[hex] 4888 8A82
6	Ok	160 µs	480 µs	102	00	24243	00	Left Wing	[hex] 8909 0C42
7	Ok	530 µs	850 µs	103	10	26263	10	Right Wing	[hex] 4989 8EC2
8	Parity Error	900 µs	1.22 ms	104	00	24683	10		[hex] 491A 0C22

Table 13-19: Content of the Decode results table

Column	Description
State	Overall state of the frame.
Start	Time of word start in relation to the trigger point
Stop	Time of word stop in relation to the trigger point
Label	The value of the label bytes
SDI	The state of the SDI bits
DATA	All 32 bits of the word.
SSM	The state of the SSM bits
Label Name	The label name
Data (32 bit)	The value of the data bytes

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands are described in Chapter 24.17.14.3, "Decode results", on page 1989.

13.13.5 Search on decoded ARINC 429 data

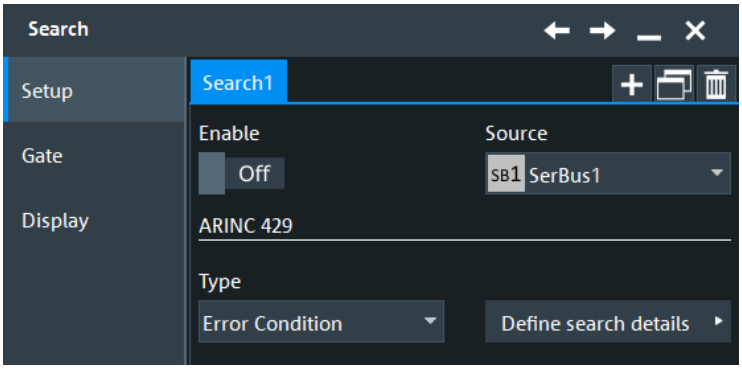
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 11, "Search functions", on page 419.

13.13.5.1 ARINC 429 search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"



Type
The search criterion is defined by "Type". All trigger types are also available for search.

Additional search parameters are provided under "Detailed Search Parameter Setup".

"Word Start" Searches for the start word.

"Word Stop" Searches for the stop word.

"Label + Data" Searches for a defined word format. You can search for the label, the data, the SDI, and SSM bits separately.
For details, see "Label + Data" on page 708.

"Error condition"
Identifies various errors in the frame, see "Error Conditions" on page 709.

Remote command:
`SEARch:TRIGger:ARINc:TYPE` on page 1993

Label setup: Condition, Label min, Label max
The label setup consists of the condition and one or two label patterns.

The label setup settings are the same as in the ARINC trigger setup, see "Label setup: Condition, Label Min, Label Max" on page 708.

Remote command:
`SEARch:TRIGger:ARINc:LABel:CONDition` on page 1994
`SEARch:TRIGger:ARINc:LABel:MIN` on page 1994
`SEARch:TRIGger:ARINc:LABel:MAX` on page 1994

Data setup: Condition, Data min, Data max
The data setup consists of the condition and one or two data patterns.

The data setup settings are the same as in the ARINC trigger setup, see "Data setup: Condition, Data Min, Data Max" on page 709.

Remote command:
`SEARch:TRIGger:ARINc:DATA:CONDition` on page 1994
`SEARch:TRIGger:ARINc:DATA:MIN` on page 1994
`SEARch:TRIGger:ARINc:DATA:MAX` on page 1994

SDI / SSM setup: SDI, SSM
The SDI / SSM setup consists of the SDI and SSM.

The SDI / SSM setup settings are the same as in the ARINC trigger setup, see "SDI / SSM" on page 709.

Remote command:
`SEARch:TRIGger:ARINc:SDI` on page 1995
`SEARch:TRIGger:ARINc:SSM` on page 1995

Error Condition
Selects the error type to be searched for. You can select one or more error types as search condition.

The error types are the same as in the ARINC trigger setup, see "Error Conditions" on page 709

Remote command:
`SEARch:TRIGger:ARINc:ERRor:CODing` on page 1995
`SEARch:TRIGger:ARINc:ERRor:PARity` on page 1995
`SEARch:TRIGger:ARINc:ERRor:TIMing` on page 1996

13.13.5.2 ARINC search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands:

- `SEARch:RESult:ARINc:WCOunt?` on page 1998
- `SEARch:RESult:ARINc:WORD<m>:DATA?` on page 1997
- `SEARch:RESult:ARINc:WORD<m>:LABel?` on page 1996
- `SEARch:RESult:ARINc:WORD<m>:PATTern?` on page 1996
- `SEARch:RESult:ARINc:WORD<m>:SDI?` on page 1998
- `SEARch:RESult:ARINc:WORD<m>:SSM?` on page 1997
- `SEARch:RESult:ARINc:WORD<m>:STARt?` on page 1999
- `SEARch:RESult:ARINc:WORD<m>:STATe?` on page 1998
- `SEARch:RESult:ARINc:WORD<m>:STOP?` on page 1998

- SEARch:RESult:ARINc:WORD<m>:SYMBol? on page 1997

13.14 SpaceWire (option R&S RTO6-K530)

The SpaceWire is a communication network standard used for spacecrafts. It is based on the IEEE 1355 standard of comunications and coordinated by the European Space Agency (ESA).

- SpaceWire basics 714
- SpaceWire configuration 715
- SpaceWire trigger 719
- SpaceWire decode results 721
- Search on decoded SpaceWire data 723

13.14.1 SpaceWire basics

The SpaceWire links are a Point-to-point (P2P) connection between a node and another node or a rooter. The link is full-duplex bidirectional serial data link.

The SpaceWire has two types of characters:

- Data characters containing a parity bit, a data control flag and eight bits of data.

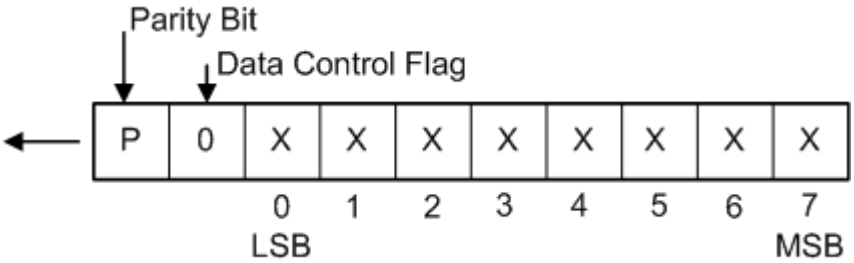


Figure 13-26: SpaceWire data characters

- Control characters containing a parity-bit, a data-control flag and the 2-bit control code. The data control flag is set to 1 and indicates that this is a control character.

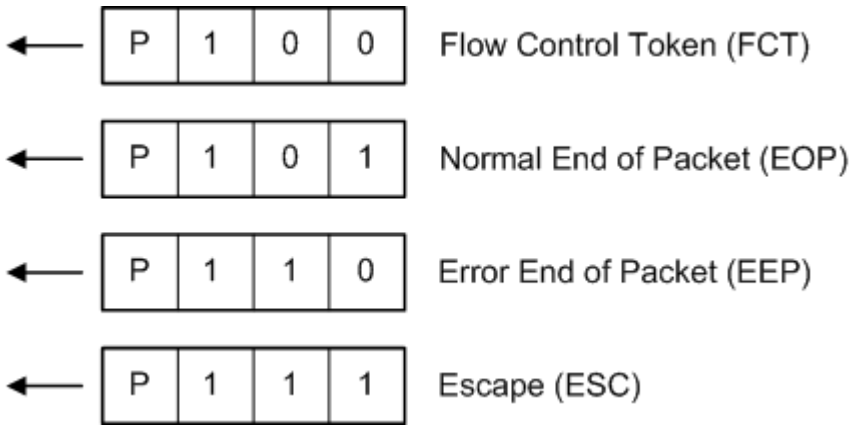


Figure 13-27: SpaceWire control characters

Also there are two control codes:

- NULL code consisting of an Escape (ESC) and a Flow Control Token (FCT)
- Time Code consisting of an ESC followed by a single data character

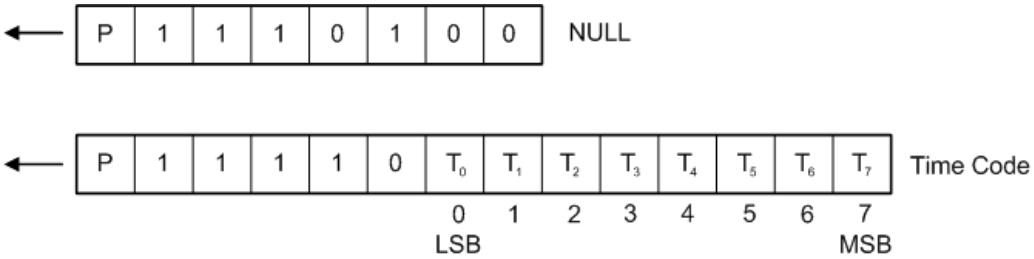


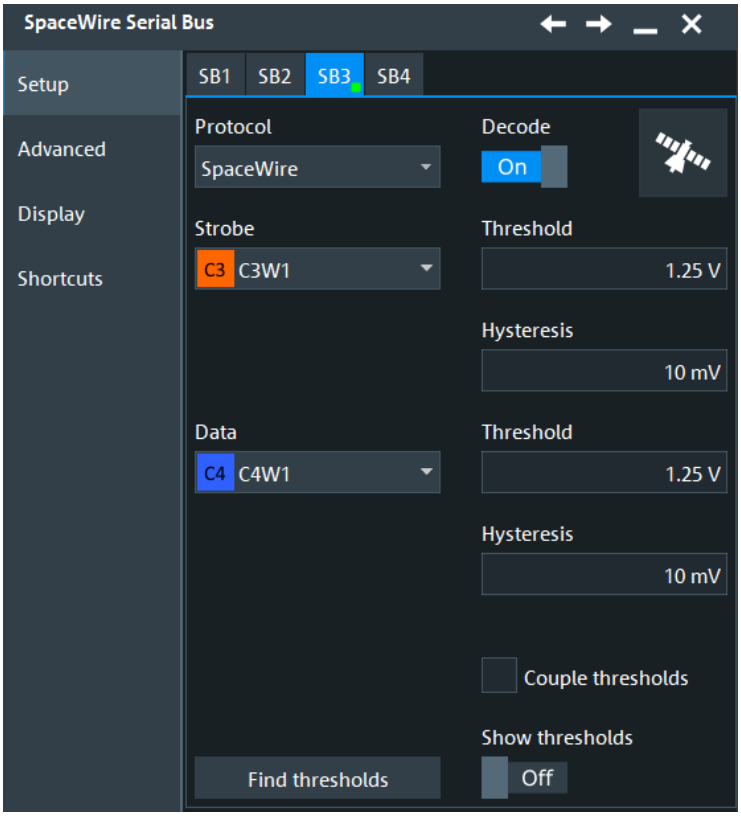
Figure 13-28: SpaceWire control codes

13.14.2 SpaceWire configuration

13.14.2.1 SpaceWire configuration settings

Access: [Bus] > "SpaceWire" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Strobe
Selects the source for the strobe signal.

Remote command:
`BUS<m>:SWIRe:STRBe:SOURce` on page 2001

Data
Selects the source for the data signal.

Remote command:
`BUS<m>:SWIRe:DATA:SOURce` on page 2000

Threshold setup
Sets the threshold value for the strobe/data signal.

"Threshold"	Enter individual values for each line directly in the fields.
"Couple thresholds"	Enables coupling, i.e. the same threshold and hysteresis value is used for the strobe and data signal.

Remote command:
`BUS<m>:SWIRe:COUPling` on page 2002
`BUS<m>:SWIRe:STRBe:THReshold` on page 2001
`BUS<m>:SWIRe:DATA:THReshold` on page 2000
`BUS<m>:SWIRe:PRESet` on page 2002
`BUS<m>:SETReflevels` on page 1671

Hysteresis
Sets a value for the hysteresis of the strobe/data signal.

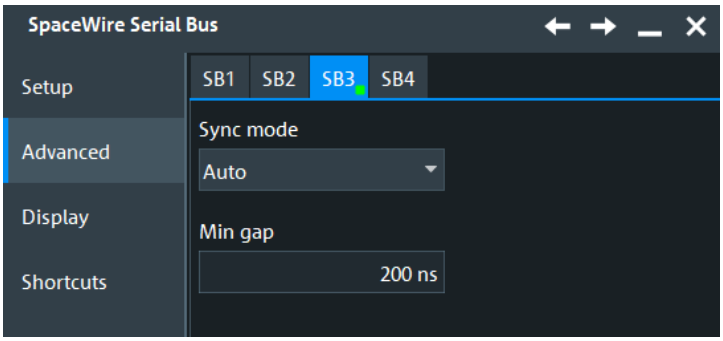
Remote command:
`BUS<m>:SWIRe:STRBe:HYSTeresis` on page 2001
`BUS<m>:SWIRe:DATA:HYSTeresis` on page 2000

Show thresholds
If enabled, the threshold levels are displayed in the diagram.

Remote command:
`BUS<m>:THReshold` on page 1671

13.14.2.2 Advanced settings

Access: [Bus] > "Advanced" tab.



Bit Position
Sets the bit position, the align position for the manual synchronization mode. This can be useful when parity errors exist in the signal, and parity check is the main indicator for the decoder to do packet alignment.

Remote command:
`BUS<m>:SWIRe:BPOSition` on page 1999

Min Gap
SpaceWire can have idle phases where strobe and data signals are not being sent. These "gaps" are identified to resume decoding after this idle time.

"Min Gap" sets the minimum duration of a gap. Any inactivity greater than this time is interpreted as a gap and lead to a resynchronization to the signal.

Remote command:
`BUS<m>:SWIRe:MGAP` on page 2000

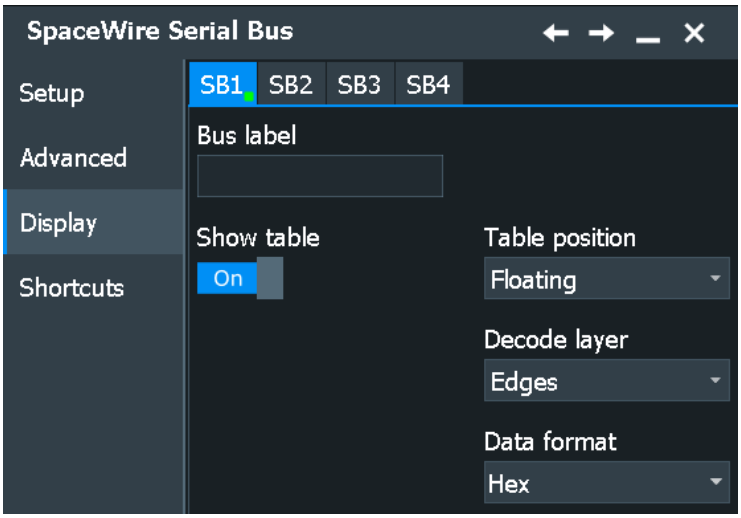
Sync Settings
Sets the mode for the synchronization of the signal. In the auto mode, the decoder automatically does the packet align. In the manual mode, you can set the align point manually with the "Bit Position" setting.

Remote command:
`BUS<m>:SWIRe:SYSLect` on page 2002

13.14.2.3 Display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.



Decode Layer
Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

Available are:

- "Off"
- "Edges"
- "Binary"

13.14.2.4 Configuring the SpaceWire signals

For configuration, assign the lines to the input channels and define the logical thresholds and the hysteresis.

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. In the "Setup" tab, set "Protocol"="SpaceWire".
4. Select the source and polarity for the strobe and data signals.
5. Enter the "Threshold" and the "Hysteresis" for the strobe and data signals.
6. Set the "Sync Settings" and the "Bit Position" if necessary.
7. Enable "Decode".

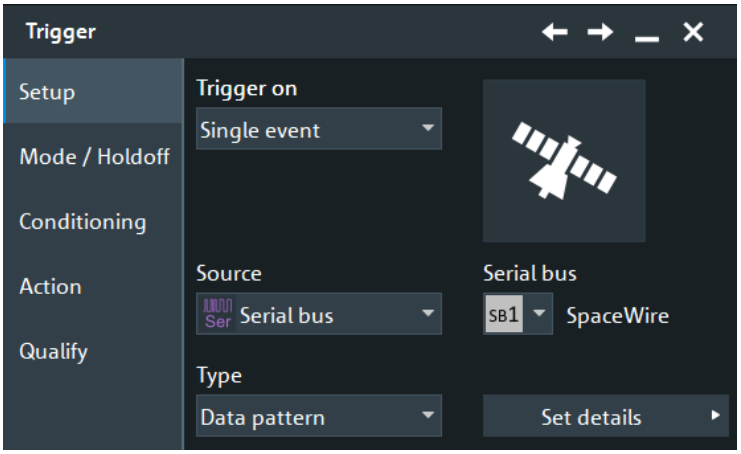
13.14.3 SpaceWire trigger

13.14.3.1 SpaceWire trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"

Make sure that:

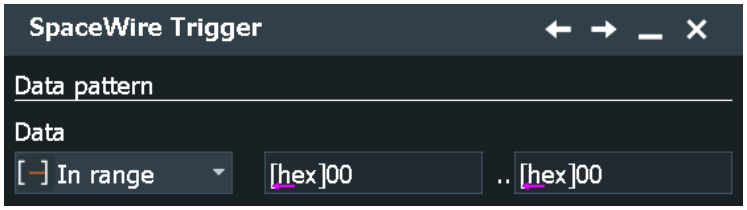
- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.



Type
Selects the trigger type for the SpaceWire analysis.

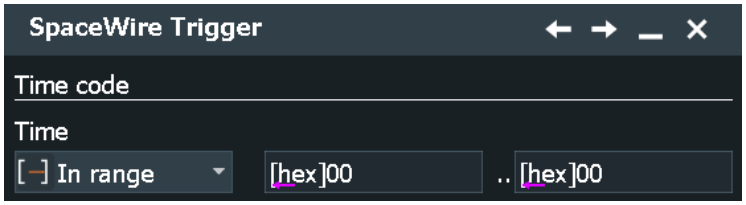
"Control frame" Sets the trigger to the selected control type frame.

"Data Pattern" Sets the trigger to a defined data pattern or pattern range.



"NULL frame" Sets the trigger to a null frame, a frame without usable data.

"Time Code" Sets the trigger to a time-code control code. You can define the data pattern of the time code to be triggered on.



"Errors" Triggers on an enabled error type.

Remote command:
`TRIGger<m>:SWIRe:TYPE` on page 2005

Trigger Control Type
Triggers on a specific control type character.

"ANY" Any control type character

"FCT" Flow control token character

"EOP" Normal end of packet character

"EEP" Error end of packet character

Remote command:
`TRIGger<m>:SWIRe:CTYPe` on page 2003

Data (Time Code)
Sets the specified data type for the time code to be triggered on. The trigger data setup consists of a comparison condition and one or two data patterns.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Condition" Sets the comparison condition to a specific value or a range.

"Data Min/ Data" Specifies the value or sets the start value of a range.

"Data Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:SWIRe:TIME:CONDition` on page 2004
`TRIGger<m>:SWIRe:TIME:MAX` on page 2005
`TRIGger<m>:SWIRe:TIME:MIN` on page 2005

Data (Data Pattern)
Sets the specified data type for the data pattern to be triggered on. The data type setup consists of the condition and one or two data patterns.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Condition" Sets the comparison condition to a specific value or a range.

"Data Min/ Data" Specifies the value or sets the start value of a range.

"Data Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:SWIRe:DATA:CONDition` on page 2003
`TRIGger<m>:SWIRe:DATA:MAX` on page 2003
`TRIGger<m>:SWIRe:DATA:MIN` on page 2004

Parity Error
Checks the parity of every frame and triggers if the parity is even.

Remote command:
`TRIGger<m>:SWIRe:ERRor:PARity` on page 2004

ESC Error
Triggers on an escape error.

Remote command:
`TRIGger<m>:SWIRe:ERRor:ESC` on page 2004

13.14.3.2 Triggering on SpaceWire

Prerequisite: A bus is configured for the SpaceWire signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [Bus] key and select the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to SpaceWire.
5. Select the "Trigger Type" to be used for SpaceWire protocol analysis.
6. To refine the trigger settings, configure additional settings, which are available for some trigger types.
   For details, see Chapter 13.14.3.1, "SpaceWire trigger settings", on page 719.

13.14.4 SpaceWire decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode results table

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

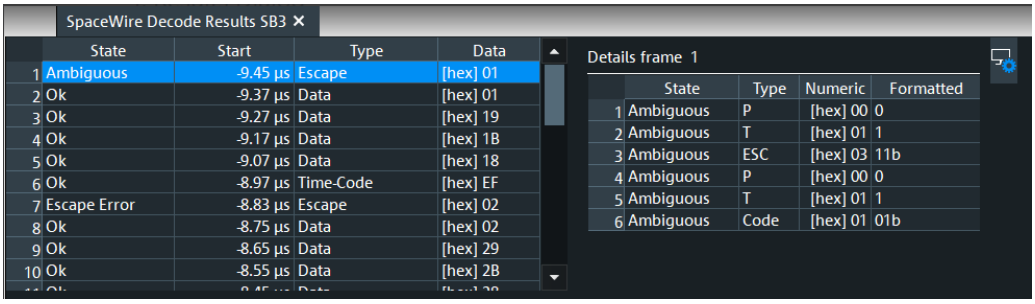


Table 13-20: Content of the Decode results table

Column	Description
State	Overall state of the frame
Start	Time of frame start in relation to the trigger point
Stop	Time of frame stop in relation to the trigger point
Type	Frame type
Data	Data value

Table 13-21: Content of the Details frame table

Column	Description
State	State of the field
Type	Field type
Numeric	Hexadecimal value of the field
Formatted	Formatted content of the field

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands to retrieve decode results are described in Chapter 24.17.15.3, "Decode results", on page 2005.

13.14.5 Search on decoded SpaceWire data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

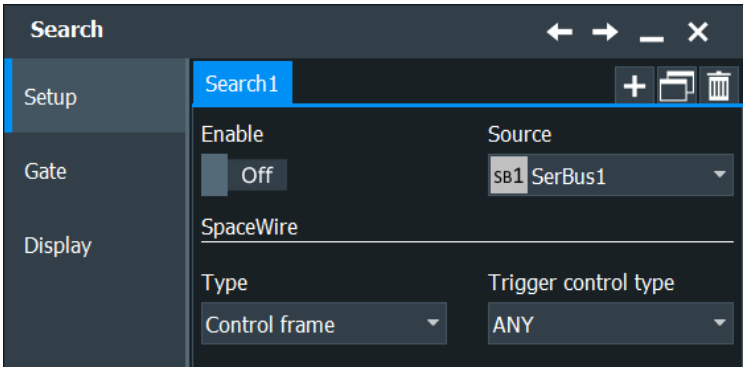
Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 11, "Search functions", on page 419.

13.14.5.1 SpaceWire search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"



Type
Sets the search type for the SpaceWire analysis.

"Control frame"	Searches for the selected control type frame.
"Data Pattern"	Searches for a defined data pattern or pattern range.
"NULL frame"	Searches for a null frame, a frame without usable data.
"Time Code"	Searches for a time-code control code. You can define the data pattern of the time code to be searched for.
"ERRORs"	Searches for an enabled error type.

Remote command:
`SEARch:TRIGger:SWIRe:TYPE` on page 2014

Trigger Control Type
Searches for a specific control type character.

"ANY"	Any control type character
"FCT"	Flow Control Token character
"EOP"	Normal End of Packet character
"EEP"	Error End of Packet character

Remote command:
`SEARch:TRIGger:SWIRe:CTYPe` on page 2011

Data (Time Code)
Sets the specified data type for the time code to be searched for. The data type setup consists of the condition and one or two data patterns.

"Condition"	Defines the operator to set a specific data type ("Equal" or "Not equal") or a data type range.
"Data Min/ Data"	Defines the bit pattern of the data pattern. In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.
"Data Max"	The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:SWIRe:TIME:CONDition` on page 2013
`SEARch:TRIGger:SWIRe:TIME:MAX` on page 2013
`SEARch:TRIGger:SWIRe:TIME:MIN` on page 2013

Data (Data Pattern)
Sets the specified data type for the data pattern to be searched for. The data type setup consists of the condition and one or two data patterns.

"Condition"	Defines the operator to set a specific data type ("Equal" or "Not equal") or a data type range.
"Data Min/ Data"	Defines the bit pattern of the data pattern. In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.
"Data Max"	The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:SWIRe:DATA:CONDition` on page 2011
`SEARch:TRIGger:SWIRe:DATA:MAX` on page 2012
`SEARch:TRIGger:SWIRe:DATA:MIN` on page 2012

Parity Error
Checks the parity of every frame and searches for even parity.

Remote command:
`SEARch:TRIGger:SWIRe:ERRor:PARity` on page 2012

ESC Error
Searches for an escape error.

Remote command:
`SEARch:TRIGger:SWIRe:ERRor:ESC` on page 2012

13.14.5.2 SpaceWire search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands:

- `SEARch:RESult:SWIRe:FCOunt?` on page 2014
- `SEARch:RESult:SWIRe:FRAMe<m>:DATA?` on page 2014
- `SEARch:RESult:SWIRe:FRAMe<m>:STARt?` on page 2015
- `SEARch:RESult:SWIRe:FRAMe<m>:STATe?` on page 2015
- `SEARch:RESult:SWIRe:FRAMe<m>:STOP?` on page 2015
- `SEARch:RESult:SWIRe:FRAMe<m>:TYPE?` on page 2016

13.14.5.3 Searching SpaceWire

Prerequisite: A serial bus is configured for the SpaceWire signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Tap "Setup search".
3. If the dialog box does not contain a search entry, tap the "Add" icon to create one, as described in "To create a user-defined search" on page 434.
4. Tap "Source" and select the serial bus that is set to SpaceWire (e.g. "SB1", unless already selected).
5. Specify search criteria according to Chapter 13.14.5.1, "SpaceWire search setup", on page 723.
6. To acquire a waveform, press [Single].

   The R&S RTO6 performs a SpaceWire decode according to the thresholds and protocol settings of the associated serial bus source.

7. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

   The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.15 Ethernet 10BASE-T and 100BASE-TX (option R&S RTO6-K540)

Twisted-pair Ethernet technologies are based on the family of standards IEEE 802.3, issued by the Institute of Electrical and Electronics Engineers (IEEE).

R&S RTO6-K540 is a firmware option that enables the R&S RTO6 to analyze Ethernet protocol variants 10BASE-T and 100BASE-TX, by decoding the signal and searching within the decoded events. It is possible to trigger on 10BASE-T signals and 100BASE-TX signals. The option is compatible with the standards IEEE 802.3i of 1990 (10BASE-T) and IEEE 802.3u of 1995 (100BASE-TX).

- The Ethernet protocol....726
- Ethernet configuration....727
- Ethernet trigger....732
- Ethernet decode results....736
- Search on decoded Ethernet data....737

13.15.1 The Ethernet protocol

The two Ethernet protocol variants that R&S RTO6-K540 can process have the following features:

- 10BASE-T uses Manchester coding (or phase encoding, PE). In terms of a logical Boolean operation, the Manchester value of each bit is the exclusive disjunction (XOR) of the original data value and the clock value. A "0" is expressed by a high-to-low transition, a "1" by a low-to-high transition. These transitions, which occur at the middle of each bit period, make the signal self-clocked.
- 100BASE-TX uses a 4B5B multi-level transmit (MLT-3) encoding. The protocol sequentially cycles through a sequence of the voltage levels -1 V, 0 V, +1 V, and 0 V. To transmit a "1" bit, MLT-3 moves to the next state; to transmit a "0" bit, it stays in the same state. 4B5B block coding is used to map groups of four bits onto groups of five bits. Also, the signal is scrambled.

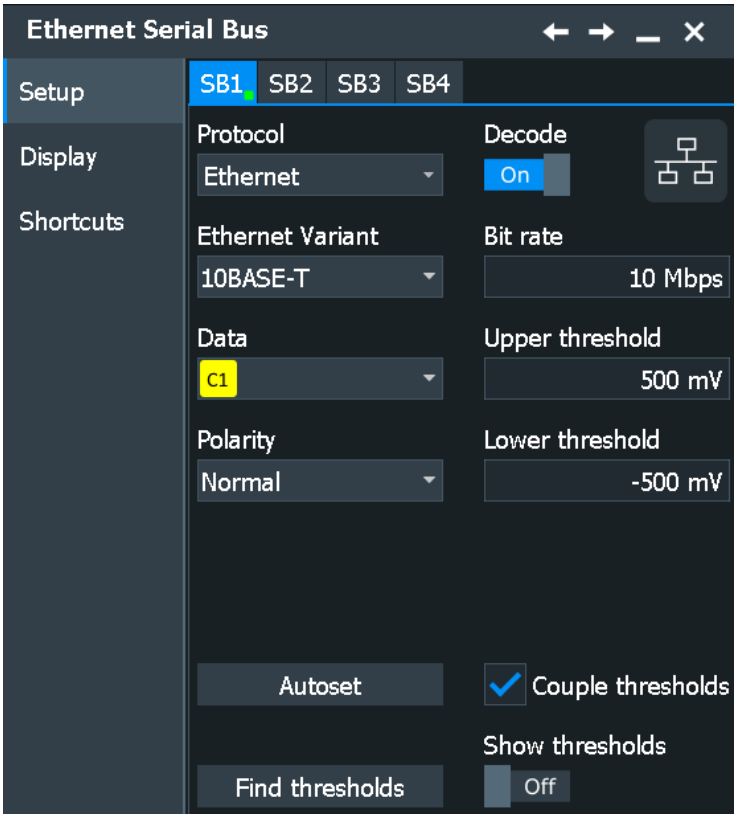
All Ethernet-over-twisted-pair technologies use wires with four twisted pairs of cables (and 8P8C connectors), but 10BASE-T and 100BASE-TX only require two pairs of wires.

13.15.2 Ethernet configuration

13.15.2.1 Ethernet configuration settings

Access: [Bus] > "Ethernet" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Protocol
Defines the protocol type of the selected serial bus.

Remote command:
`BUS<m>:TYPE` on page 1670

Decode
Enables the decoding of the selected bus. The signal icon of the bus appears on the signal bar.

Remote command:
`BUS<m>[:STATe]` on page 1670

Ethernet Variant
Defines the Ethernet protocol variant and transmission speed.

"10BASE-T"
Selects the Ethernet protocol variant 10BASE-T (standard data rate 10 Mbit/s).

"100BASE-TX"
Selects the Ethernet protocol variant 100BASE-TX, which provides 100 Mbit/s use data rate. Due to 4b/5b encoding, the raw data rate on the line is 125 Mbit/s. This value is used by R&S RTO6-K540 as the bit rate default for 100BASE-TX.

Remote command:
`BUS<m>:ETHernet:VARiant` on page 2017

Source
Defines the source settings for the data signal.

Permitted source selections are the analog, mathematical, and reference channels.

As soon as the serial bus trigger has been selected, the only permitted source selections are the analog channels "C1" – "C4", which are required for triggering.

Remote command:
`BUS<m>:ETHernet:SOURce` on page 2017

Polarity
Defines the polarity ("Normal" or "Inverted") of the data signal. This setting is only available in 10BASE-T.

Remote command:
`BUS<m>:ETHernet:POLarity` on page 2017

Thresholds
Sets the threshold value for the digitization of each signal line. If the signal value on the line is higher than the upper threshold, the signal state is high. Otherwise, if the signal value is below the lower threshold, the signal state is considered low.

"Thresholds" Enter the values directly: upper threshold in the upper field, lower threshold in the lower field.

"Autoset" Starts software algorithms for determining the signal threshold levels and bitrate.

"Find thresholds" Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

"Couple thresholds" Couples threshold settings between upper and lower threshold.

Remote command:
`BUS<m>:ETHernet:THReshold:HIGH` on page 2018
`BUS<m>:ETHernet:THReshold:LOW` on page 2018
`BUS<m>:ETHernet:PRESet` on page 2018
`BUS<m>:SETReflevels` on page 1671
`BUS<m>:FAUToset` on page 1671

Bit rate
Defines the transmission speed setting for the data signal:

- 10BASE-T: default bit rate 10 Mbps
- 100BASE-TX: default bit rate 125 Mbps

In both variants, the permitted bit rates range from 10 kbps to 150 Mbps. Switching the variant adjusts the bit rate, independent of the previous setting.

Remote command:
`BUS<m>:ETHernet:BITRate` on page 2019

Show thresholds
If enabled, the threshold levels are displayed in the diagram.

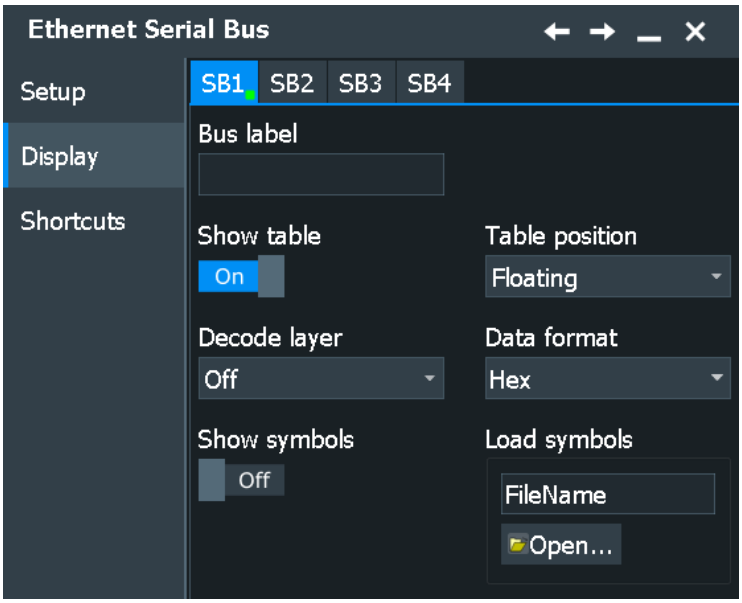
Remote command:
`BUS<m>:THReshold` on page 1671

13.15.2.2 Ethernet display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

To enhance the decode possibilities of the Ethernet protocol, you can use an additional setting in the "Display" tab: "Decode layer".



Ethernet Specific Label Settings

Label lists are protocol-specific. An Ethernet label file contains two values for each identifier:

- "Ethernet Header Display": the Ethernet header display value (e.g. `FF:FF:FF:FF:FF:FF/48`)
- "Symbolic label": user-defined symbolic name of the header, specifying its function (e.g. `BroadCast`).

For example, an ethernet label list can be a vendor-specific MAC address list:

```
# ---------------------------------------------------------------------------
@FILE_VERSION  = 1.00
@PROTOCOL_NAME = ethernet
# ---------------------------------------------------------------------------
# Labels for Ethernet protocol
#   Column order: Ethernet Header Display, Label
#---------------------------------------------------------------------------
# Supported MAC Address Format
# xx:xx:xx  standard 24 bit manufactory header
# xx:xx:xx:xx:xx:xx/yy  support other length headers
# yy should be the header length in decimal
# yy should be between 24 - 48
# ---------------------------------------------------------------------------
00:00:0C, Cisco
00:01:13, Olympus
00:01:14, KandaTsu
00:04:07, TopconPo
00:0B:64, KiebackP
00:1B:C5:06:C0:00/36, LuxconSy
00:1B:C5:06:D0:00/36, TesElect
00:1B:C5:06:E0:00/36, TwoDimen
00:1B:C5:06:F0:00/36, LlcEmzio
00:1F:BE, Shenzhen
00:21:8F, Avantgar
00:21:90, GoliathS
00:21:91, D-Link
00:21:92, BaodingG
00:50:C2:5F:60:00/36, Cambridg
00:50:C2:5F:70:00/36, Metrolog
00:50:C2:5F:80:00/36, GrupoEpe
18:42:2F, AlcatelL
18:44:62, RiavaNet
18:46:17, SamsungE
18:8E:D5, TpVision
18:E7:F4, Apple
40:D8:55:1C:80:00/36, SensataT
40:D8:55:1C:90:00/36, Andy-L
40:D8:55:1C:A0:00/36, RigelEng
40:D8:55:1C:B0:00/36, MgSRL
40:D8:55:1C:D0:00/36, YxlonInt
40:D8:55:1C:E0:00/36, PeterHub
40:D8:55:1C:F0:00/36, OmnikNew
40:D8:55:1D:00:00/36, WebeasyB
FC:F8:B7, TronteqE
FC:FA:F7, Shanghai
FC:FB:FB, Cisco
```

```
FC:FE:77, HitachiR
FF:FF:FF:FF:FF:FF/48, BroadCast
```

In the figure below, you can see a decoded broadcast symbol. The symbol name is displayed in the "Destination symbol" column of the "Decode restuls" table.

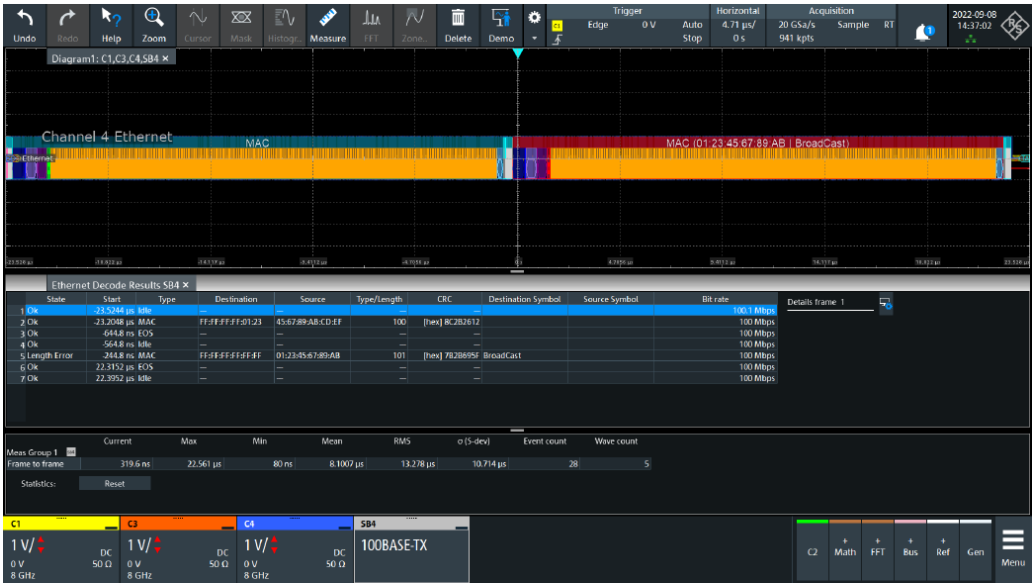


Figure 13-29: Decoded Ethernet results with applied label

Decode layer

Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

Available are:

- "Edges": Shows where the decoder identified transitions through the threshold.
- "Binary": Shows the bits as identified by the decoder.
- "Descramble": Shows the bits after signal has been descrambled.
- "5b Symbol": Shows the symbol values after the 4b/5b decoder.

13.15.2.3 Configuring Ethernet signals

For configuration, assign the lines to the input channels and define the active states and the logical thresholds.

For details on configuration settings, see Chapter 13.15.2.1, "Ethernet configuration settings", on page 727.

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. In the "Setup" tab, select the protocol: "Ethernet".

4. Tap "Ethernet Variant" and select the variant ("10BASE-T" or "100BASE-TX") you want to set up.

   Note: Note that no triggering on the serial bus is available.
   To trigger the signal, use the edge trigger on the source channel.

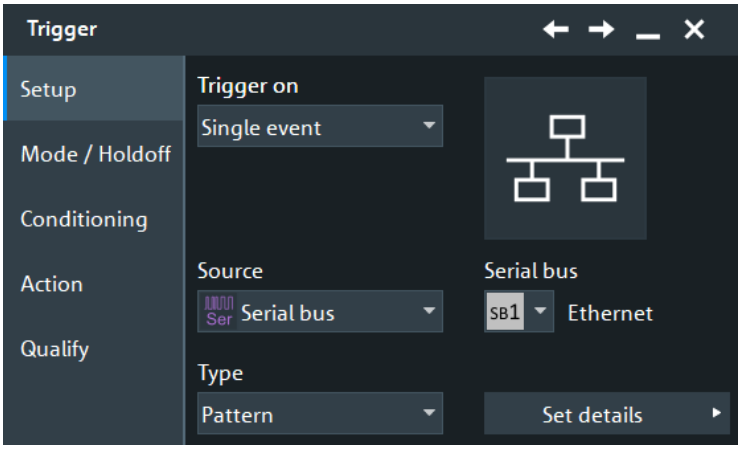
5. For the variant "10BASE-T", select the polarity ("Normal" or "Inverted") of the data signal.
6. Set the logical thresholds.
7. Enable "Decode".

13.15.3 Ethernet trigger

If you need information on how to get started with triggering on Ethernet 10BASE-T signals, see Chapter 13.15.3.2, "Triggering on Ethernet signals", on page 736. Otherwise proceed with the Ethernet trigger settings.

13.15.3.1 Ethernet trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



Make sure that:

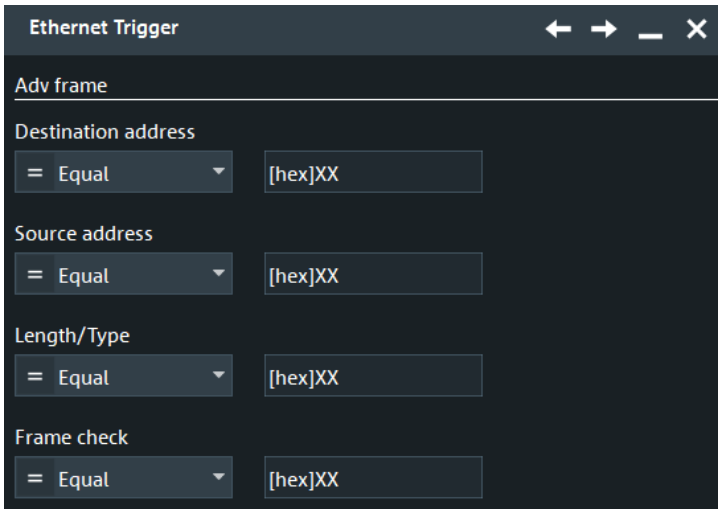
- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Type
Selects the trigger type for the Ethernet analysis.

"Frame Start" Available only for "Variant > 10BASE-T".
Sets the trigger to the start frame delimiter (SFD). The start of frame (SOF) condition is the occurrence of the preamble; the trigger instant is after the SFD.

"Pattern" Available only for "Variant > 10BASE-T".
Sets the trigger to any bit pattern (data) that can be freely specified, starting from the beginning of the frame. The trigger instant is after the last bit of the specified data pattern.

"Adv frame " Triggers on an advanced frame.



"Adv error" Triggers on any selected errors.

Remote command:
`TRIGger<m>:ETHernet:TYPE` on page 2020

Ethernet Variant
Defines the Ethernet protocol variant and transmission speed.

"10BASE-T"
Selects the Ethernet protocol variant 10BASE-T (standard data rate 10 Mbit/s).

"100BASE-TX"
Selects the Ethernet protocol variant 100BASE-TX, which provides 100 Mbit/s use data rate. Due to 4b/5b encoding, the raw data rate on the line is 125 Mbit/s. This value is used by R&S RTO6-K540 as the bit rate default for 100BASE-TX.

Remote command:
`BUS<m>:ETHernet:VARiant` on page 2017

Data Pattern
Specifies the data pattern that is to be triggered on.

Remote command:
`TRIGger<m>:ETHernet:PATTern` on page 2020

Bit rate
Defines the transmission speed setting for the data signal:

- 10BASE-T: default bit rate 10 Mbps
- 100BASE-TX: default bit rate 125 Mbps

In both variants, the permitted bit rates range from 10 kbps to 150 Mbps. Switching the variant adjusts the bit rate, independent of the previous setting.

Remote command:
`BUS<m>:ETHernet:BITRate` on page 2019

Bit index
Defines the position of the first bit of the data pattern.

"Bit index operator" Sets the operator ("Equal", "Greater or equal", or "In range").

"Bit index" Sets the bit index (data position), or the start value of a bit index range.

"Bit index to" Sets end value of a bit index range (data position range). Available only, if the "Bit index operator" is set to "In range".

Destination Address
Defines the destination address.

"Condition" Sets the comparison condition to a specific value or a range.

"Destinataion Address (Min)" Specifies the value or sets the start value of a range.

"Destinataion Adress (Max)" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:ETHernet:FRAMe:DCONdition` on page 2022
`TRIGger<m>:ETHernet:FRAMe:DMAX` on page 2022
`TRIGger<m>:ETHernet:FRAMe:DMIN` on page 2023

Source Address
Defines the source address, the physical address of the device that sends the frame.

"Condition" Sets the comparison condition to a specific value or a range.

" Source Address (Min)" Specifies the value or sets the start value of a range.

"Source Adress (Max)" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:ETHernet:FRAMe:SCONdition` on page 2023
`TRIGger<m>:ETHernet:FRAMe:SMAX` on page 2023
`TRIGger<m>:ETHernet:FRAMe:SMIN` on page 2023

Length/ Type
Defines the length/type field value.

"Condition" Sets the operator to trigger on a specific address pattern ("Equal" or "Not equal") or an address range.

"Length/Type (Min)" Defines the length/type pattern for all operators that require one pattern.

"Length/Typ (Max)" Defines the length/type pattern that is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:ETHernet:FRAMe:TCONdition` on page 2024
`TRIGger<m>:ETHernet:FRAMe:TMAX` on page 2024
`TRIGger<m>:ETHernet:FRAMe:TMIN` on page 2024

Frame check
Defines the frame check sequence value.

"Condition" Sets the comparison condition to a specific value or a range.

"Frame check (Min)" Specifies the value or sets the start value of a range.

"Frame check (Max)" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:ETHernet:FRAMe:CCONdition` on page 2021
`TRIGger<m>:ETHernet:FRAMe:CMAX` on page 2022
`TRIGger<m>:ETHernet:FRAMe:CMIN` on page 2022

Error
Triggers on enabled errors.

"Preamble error" Triggers on a frame with invalid preamble.

"Length error" Triggers on an incorrect length of the sequence - when additional or missing bits are detected and the sequence of bits is not as expected.

"CRC error" Triggers on a frame that has a mismatch of the cyclic redundancy check (CRC) value between the transmitting and receiving device.

Remote command:
`TRIGger<m>:ETHernet:ERRor:CRC` on page 2020
`TRIGger<m>:ETHernet:ERRor:LENGth` on page 2021
`TRIGger<m>:ETHernet:ERRor:PREamble` on page 2021

13.15.3.2 Triggering on Ethernet signals

Prerequisite: A bus is configured for the Ethernet signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to Ethernet.
5. Tap "Type" and select the trigger type to be used for Ethernet protocol analysis. Available trigger types are "Frame Start", "Pattern", "Adv frame" and "Adv error".
6. Depending on the selected Ethernet variant, more setup conditions have to be specified.

For information on how to proceed with the configuration settings, see Chapter 13.15.3.1, "Ethernet trigger settings", on page 732.

13.15.4 Ethernet decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

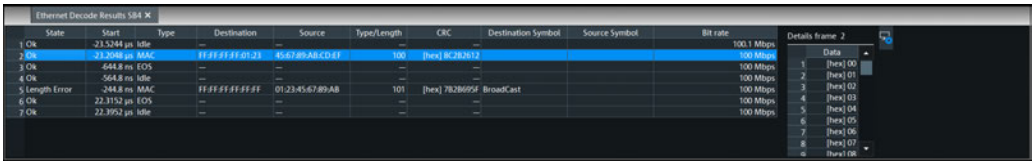


Table 13-22: Content of the Decode results table

Column	Description
State	Overall state of the frame: either OK or the relevant error condition (preamble, length)
Start	Start time of the frame
Type	Frame type
Destination	Destination address of the frame
Source	Source address of the frame
Type/Length	The subprotocol (e.g. HTML, video, etc.) determines what meaning this field has. Since the content of this data area is not decoded, the interpretation of this field is ambivalent. It could either be the word type (specific for the subprotocol) or the word length.
Data	Values of the data bytes in a frame. The table shows a truncated version; to see all the bytes in a separate data table, activate "Show details". The data format is always hexadecimal.
CRC	FrameCheck (Cyclic Redundancy Code, CRC)
Destination Symbol	Translation (or symbolic label) of the destination address, if the label list is enabled.
Source Symbol	Translation (or symbolic label) of the source address, if the label list is enabled.
Number of Words	Number of words in the frame
Bit rate	Value of the bit rate

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands are described in Chapter 24.17.16.3, "Decode results", on page 2024.

13.15.5 Search on decoded Ethernet data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

Before you can start the search, you have to configure the bus correctly and acquire decoded data.

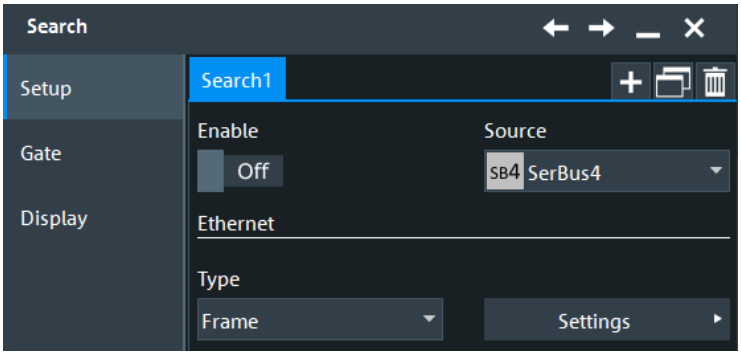
To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 11, "Search functions", on page 419.

If you need information on how to get started with searching Ethernet data, see Chapter 13.15.5.3, "Searching Ethernet data", on page 741. Otherwise proceed with the Ethernet search setup.

13.15.5.1 Ethernet search setup

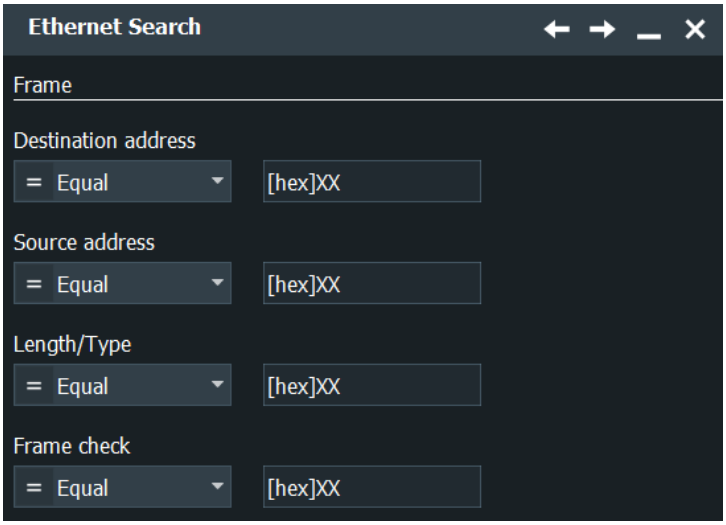
Access: [Bus] > "Shortcuts" tab > "Setup search"



Type
Define the event types to be searched. Available event types are "Frame" and "Error".

Remote command:
`SEARch:TRIGger:ETHernet:TYPE` on page 2030

Frame
Searches for the following frame conditions: "Destination address", "Source address", "Length/Type", or "Frame check".



Destination address ← Frame
To search for a destination address, an address pattern or optionally an address range have to be specified.

Remote command:
`SEARch:TRIGger:ETHernet:FRAMe:DCONdition` on page 2031
`SEARch:TRIGger:ETHernet:FRAMe:DMIN` on page 2031
`SEARch:TRIGger:ETHernet:FRAMe:DMAX` on page 2031

Source address ← Frame
To search for a source address, an address pattern or optionally an address range have to be specified.

Remote command:
`SEARch:TRIGger:ETHernet:FRAMe:SCONdition` on page 2032
`SEARch:TRIGger:ETHernet:FRAMe:SMIN` on page 2032
`SEARch:TRIGger:ETHernet:FRAMe:SMAX` on page 2032

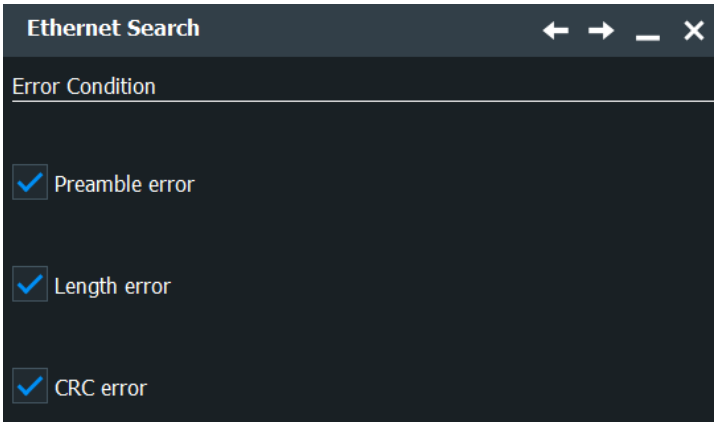
Length/Type ← Frame
To search for a frame length or frame type, a type/length pattern or optionally a range of type/length patterns have to be specified.

Remote command:
`SEARch:TRIGger:ETHernet:FRAMe:TCONdition` on page 2033
`SEARch:TRIGger:ETHernet:FRAMe:TMIN` on page 2033
`SEARch:TRIGger:ETHernet:FRAMe:TMAX` on page 2033

Frame check ← Frame
To search for a specific pattern, this pattern or optionally a range of patterns have to be specified.

Remote command:
`SEARch:TRIGger:ETHernet:FRAMe:CCONdition` on page 2034
`SEARch:TRIGger:ETHernet:FRAMe:CMIN` on page 2034
`SEARch:TRIGger:ETHernet:FRAMe:CMAX` on page 2034

Error Condition
Searches for the following error conditions: "Preamble Error" or "Length Error".



Preamble Error ← Error Condition
Searches for any preamble errors.

Remote command:
`SEARch:TRIGger:ETHernet:ERRor:PREamble` on page 2035

Length Error ← Error Condition
Searches for any length errors.

Remote command:
`SEARch:TRIGger:ETHernet:ERRor:LENGth` on page 2035

13.15.5.2 Ethernet search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands:

- `SEARch:RESult:ETHernet:WCOunt?` on page 2035
- `SEARch:RESult:ETHernet:WORD<m>:TYPE?` on page 2037
- `SEARch:RESult:ETHernet:WORD<m>:FTYPe?` on page 2038
- `SEARch:RESult:ETHernet:WORD<m>:STATe?` on page 2036
- `SEARch:RESult:ETHernet:WORD<m>:STARt?` on page 2036
- `SEARch:RESult:ETHernet:WORD<m>:STOP?` on page 2037
- `SEARch:RESult:ETHernet:WORD<m>:DESTaddress?` on page 2037
- `SEARch:RESult:ETHernet:WORD<m>:SRCaddress?` on page 2037

- `SEARch:RESult:ETHernet:WORD<m>:DATA?` on page 2038
- `SEARch:RESult:ETHernet:WORD<m>:CRC?` on page 2038
- `SEARch:RESult:ETHernet:WORD<m>:DSYMbol?` on page 2039
- `SEARch:RESult:ETHernet:WORD<m>:SSYMbol?` on page 2039
- `SEARch:RESult:ETHernet:WORD<m>:BYTE<n>:VALue?` on page 2039

13.15.5.3 Searching Ethernet data

Prerequisite: A serial bus is configured for the Ethernet signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Tap "Setup search".
3. If the dialog box does not contain a search entry, tap the "Add" icon to create one, as described in "To create a user-defined search" on page 434.
4. Tap "Source" and select the serial bus that is set to Ethernet (e.g. "SerBus1", unless already selected).

   The search dialog for Ethernet protocol analysis is opened.
5. Specify search criteria according to Chapter 13.15.5.1, "Ethernet search setup", on page 738.
6. To acquire a waveform, press [Single].

   The R&S RTO6 performs an Ethernet decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).
7. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

   The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.16 MDIO (option R&S RTO6-K540)

The R&S RTO6-K540 option enables the R&S RTO6 to analyse Management Data Input/Output (MDIO) protocols. The option is compatible with the Ethernet standard IEEE 802.3 (http://standards.ieee.org/findstds/standard/802.3-2012.html) and supports simplified triggering and decoding for both variants of MDIO: Clause 22 with basic addressing, and Clause 45 with advanced addressing that meets the requirements of 10 Gigabit Ethernet devices.

- The MDIO protocol....742
- MDIO configuration....743
- MDIO trigger....747
- MDIO decode results....750
- Search on decoded MDIO data....752

13.16.1 The MDIO protocol

MDIO is used for bidirectional transfer of control and status information between the physical layer entity (PHY) and the station management entities (STA).

A major application of MDIO is fault detection by interrogating registers of physical devices. Hence, MDIO serial bus visualization helps debugging new products by giving developers a quick insight into the native data on the bus without using a special decoder.

On physical level, MDIO is a clocked non-return-to-zero (NRZ) code similar to SPI. According to the Ethernet standard, the protocol defines two threshold levels, 2 V and 0.8 V, which establish a hysteresis.

On logical level, MDIO is a fairly simple protocol with a fixed word length of 64 bits. The structure of MDIO frames is shown in the following tables:

Table 13-23: MDIO frame structure according to Clause 22

	Management Frame Fields							
Frame	PRE	ST	OP	PHYAD	REGAD	TA	DATA	IDLE
WRITE	1...1	01	01	AAAAA	RRRRR	10	DDDDDDDDDDDDDDDD	Z
READ	1...1	01	10	AAAAA	RRRRR	Z0	DDDDDDDDDDDDDDDD	Z

Table 13-24: MDIO frame structure according to Clause 45

	Management Frame Fields							
Frame	PRE	ST	OP	PRTAD	DEVAD	TA	ADDRESS / DATA	IDLE
ADDRESS	1...1	00	00	PPPPP	EEEEE	10	AAAAAAAAAAAAAAAA	Z
WRITE	1...1	00	01	PPPPP	EEEEE	10	DDDDDDDDDDDDDDDD	Z
READ	1...1	00	11	PPPPP	EEEEE	Z0	DDDDDDDDDDDDDDDD	Z
POST-READ increment address	1...1	00	10	PPPPP	EEEEE	Z0	DDDDDDDDDDDDDDDD	Z

The following abbreviations are used in the tables:

- PRE = preamble, consisting of 32 logic "one" bits ("1...1")
- PRE = preamble, consisting of 32 logic "one" bits ("1...1")
- ST = start of frame code (2 bits), "01" for Clause 22, "00" for Clause 45, "0X" for any, no other options permitted
- OP = operation code or "OpCode" (2 bits). This is a frame type code specifying the type of transaction.

- PHYAD = address of a physical layer entity (in Clause 22)
- PRTAD = address of a port (in Clause 45)
- REGAD = register address within a PHY (in Clause 22)
- DEVAD = device address within a port (in Clause 45)
- TA = turnaround time, a 2-bit time spacing between REGAD/DEVAD and DATA. The turnaround provides the slave some time to answer upon a read command. TA is hard-wired even in write commands, although it is not required there.
- ADDRESS / DATA = address or payload data, 16 bits
- IDLE = A single value (high-impedance state) indicating to the Physical Medium Attachment (PMA) that there is no data to convey

Instead of a specific hardware trigger, the option R&S RTO6-K540 uses a predefined generic serial bus pattern trigger. It simply triggers on a bit pattern in the data stream. This is fast, but limited in the complexity of the conditions.

The MDIO trigger settings allow you to define the MDIO fields individually. The firmware concatenates the settings to a single search pattern that is then used by the serial bus pattern.

While this design is simple, it does not allow triggering on a data range or even inequality. This explains the much simpler structure compared to other protocols.

13.16.2 MDIO configuration

13.16.2.1 MDIO configuration settings

Access: [Bus] > "MDIO" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.

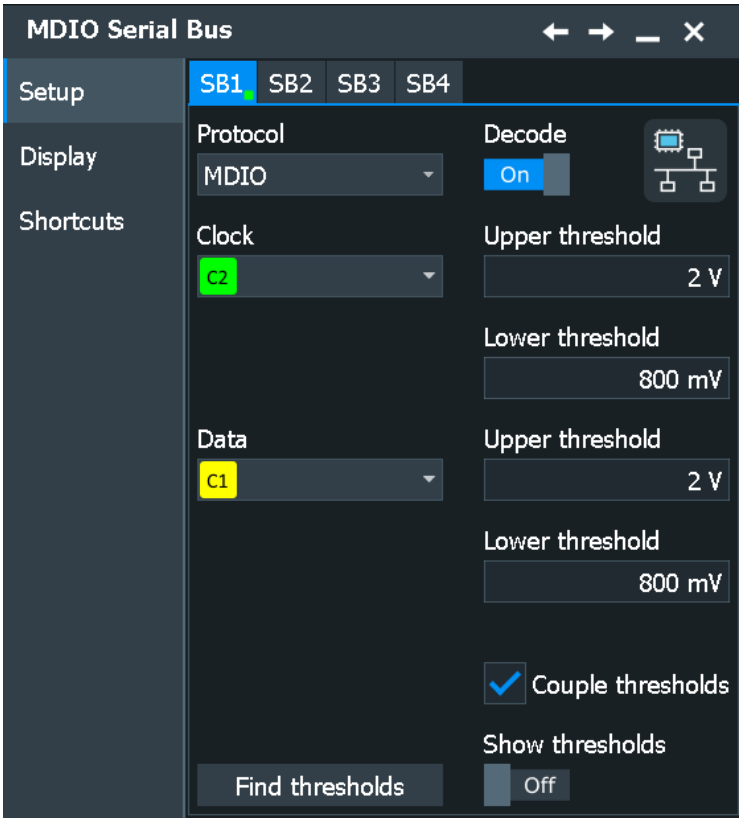


Figure 13-30: Serial bus MDIO protocol configuration dialog

See also: Chapter 13.1.1, "Setup - general settings", on page 485

Source

MDIO requires two source channels, one for clock and one for data.

"Clock" Defines the source settings for the clock line (management data clock, MDC). Typically, select any of the analog channels "C1" – "C4" or digital channels "D0" – "D15", depending on the test application. "Math" or "Ref" waveforms are also permitted.

"Data" Defines the source settings for the data signal. Typically, select any of the analog channels "C1" – "C4" or digital channels "D0" – "D15", depending on your application, but not the same as for "Clock". "Math" or "Ref" waveforms are also permitted.

Remote command:

`BUS<m>:MDIO:CLOCk:SOURce` on page 2040

`BUS<m>:MDIO:DATA:SOURce` on page 2041

Thresholds

MDIO defines two thresholds for each source line:

- Vih(min) is being used for the rising edge evaluation. This "h" (high) threshold is the minimum value for the signal to be identified as "1". If the signal value comes from a low state (hence, rising edge), the state remains to be considered as low ("0"), until it has risen above Vih(min).

- Vil(max) is being used for the falling edge evaluation. This "l" (low) threshold is the maximum level for the signal to be identified as "0". If the signal value comes from a high state (hence, falling edge), the state remains to be considered as high ("1"), until it has fallen below Vil(max).

There are four ways to set the thresholds for the digitization of the signal lines:

- "Threshold"
  Enter the values directly in the fields.
- "Preset thresholds"
  Allows to select the default threshold settings according to the Ethernet standard: 2.0 V and 0.8 V.
- "Auto thresholds"
  Executes a measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.
- "Couple thresholds"
  Overwrites the data thresholds with the clock thresholds.

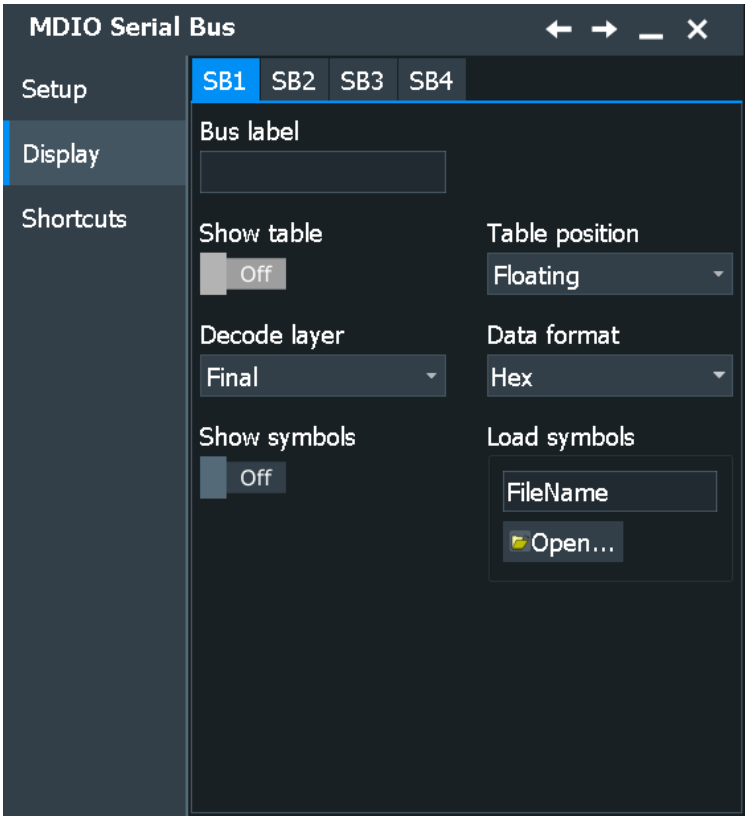
Remote command:

`BUS<m>:MDIO:CLOCk:THReshold:HIGH` on page 2041
`BUS<m>:MDIO:CLOCk:THReshold:LOW` on page 2041
`BUS<m>:MDIO:DATA:THReshold:HIGH` on page 2041
`BUS<m>:MDIO:DATA:THReshold:LOW` on page 2042
`BUS<m>:MDIO:PRESet` on page 2042
`BUS<m>:SETReflevels` on page 1671
`BUS<m>:MDIO:COUPling` on page 2042

13.16.2.2 MDIO display settings

Access: [Bus] > "Setup" tab > "Protocol = MDIO" > "Display" tab

To enhance the decode possibilities of the MDIO protocol, you can use an additional setting in the "Display" tab: "Decode layer".



Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

MDIO Specific Label Settings

Decode layer
Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

Available are:

- "Final"
- "Edges"
- "Binary"

13.16.2.3 Configuring MDIO signals

For configuration, assign the lines to the input channels and define the active states and the logical thresholds.

For details on configuration settings, see Chapter 13.16.2.1, "MDIO configuration settings", on page 743.

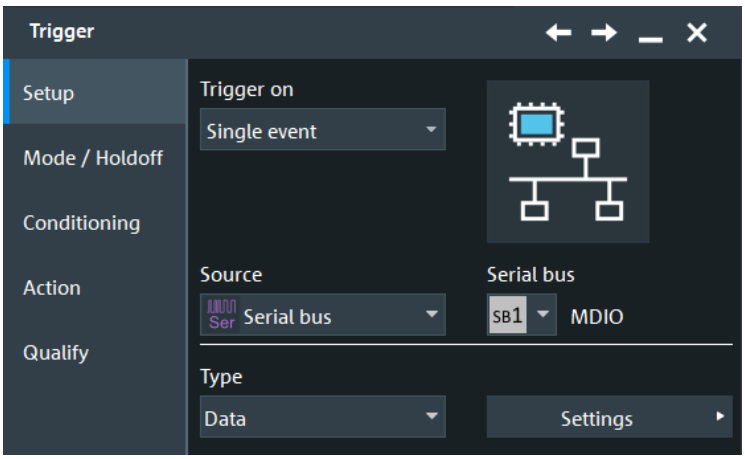
1. Press the [Bus] key on the front panel.

2. Select the tab of the bus you want to set up, for example "SB1".
3. Select the "Setup" tab.
4. In the "Setup" tab, set the "Protocol" > "MDIO".
5. Select the waveform for the "Clock" and "Data" lines.
6. Set the logical thresholds.
7. Enable "Decode".

13.16.3 MDIO trigger

13.16.3.1 MDIO trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



In this section, all trigger settings are described. The user interface of the instrument displays only appropriate settings and guides you through the trigger setup.
For a list of supported trigger conditions, refer to the data sheet.

Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Serial bus
Selects the serial bus to be triggered on. Make sure to select the correct bus before you enter the settings.

To trigger on a serial bus, the signals sources must be channel signals. If the data or clock source is a math or reference waveform, you cannot trigger on that bus.

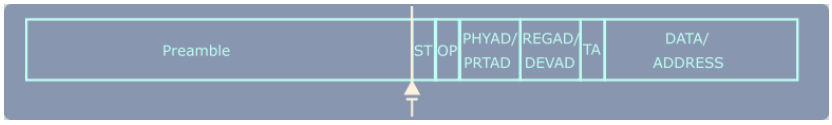
Remote command:
`TRIGger<m>:SOURce:SBSelect` on page 1675

Protocol
Defines the protocol type of the selected serial bus.

Remote command:
`BUS<m>:TYPE` on page 1670

MDIO Trigger Type
Selects the trigger type for MDIO analysis.

"Frame Start" Sets the trigger to the start of frame (SOF) field. The start of frame condition and the trigger instant is the end of the preamble.
Trigger pattern: preamble (32 bits "1")



There are no additional parameters to be specified.

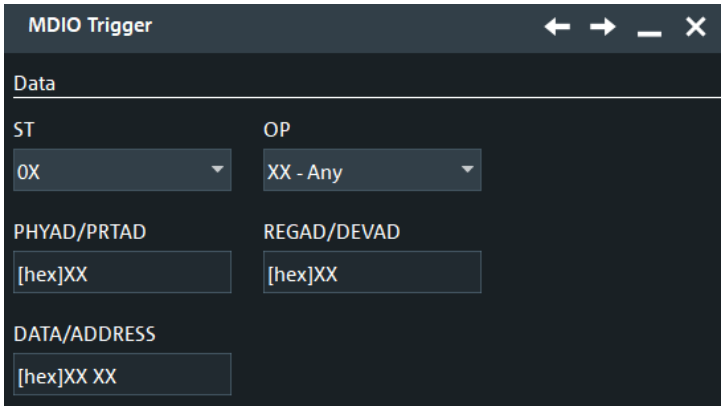
"Frame Stop" Sets the trigger to the end of frame (EOF) field. The trigger instant is after the last data bit.
Trigger pattern: preamble (32 bits "1") + 32 bits "X"

There are no additional parameters to be specified.

"Data" Sets the trigger to the data field. For more information on the data condition, see MDIO in the Ethernet standard. The trigger instant is at the end of the frame after the last data bit, as indicated in the GUI.

Note: All data triggers are always at the end of the frame, even if the specified pattern to trigger for is at a different position within the data word.

Trigger pattern: preamble (32 bits "1") + "ST" (2 bits, Start of Frame Code) + "OP" (2 bits, operation code or frame type code) + "PHYAD/PRTAD" (5 bits, Physical Layer Entity Address / Port Address) + "REGAD/DEVAD" (5 bits, Register Address / Device Address) + "TA" (2 "X" bits, turnaround time) + "DATA/ADDRESS" (16 bits)
For the parameters to be specified, see "ST" on page 749, "OP" on page 749, "PHYAD/PRTAD" on page 749, "REGAD/DEVAD" on page 750, and "DATA/ADDRESS" on page 750.



Remote command:
`TRIGger<m>:MDIO:TYPE` on page 2043

ST ← MDIO Trigger Type
Selects the start of frame code of the frame pattern; available only in trigger type "Data". Permissible frame patterns are: Clause 22, Clause 45, or Any

Remote command:
`TRIGger<m>:MDIO:ST` on page 2043

OP ← MDIO Trigger Type
Selects the type of frame code (or OP code, OpCode, operation code); available only in trigger type "Data". Available frame types are: Address, Write, Read, Post Read, or Any

Remote command:
`TRIGger<m>:MDIO:FRAMetype` on page 2044

PHYAD/PRTAD ← MDIO Trigger Type
Sets the physical address or port address (5 bits) of the frame pattern; available only in trigger type "Data".

Remote command:
`TRIGger<m>:MDIO:PHYS` on page 2044

REGAD/DEVAD ← MDIO Trigger Type
Sets the register address or device address (5 bits) of the frame pattern; available only in trigger type "Data".

Remote command:
`TRIGger<m>:MDIO:REGI` on page 2044

DATA/ADDRESS ← MDIO Trigger Type
Defines the payload data pattern or address pattern (16 bits); available only in trigger type "Data".

Remote command:
`TRIGger<m>:MDIO:DATA` on page 2045

13.16.3.2 Triggering on MDIO

Prerequisite: A serial bus is configured for the MDIO signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Tap "Serial bus" and select the serial bus that is set to MDIO.
5. Tap "Trigger Type MDIO" and select the trigger type to be used for MDIO protocol analysis.
6. If the trigger type "Data" is selected, the frame pattern has to be specified.

For information on how to proceed with the configuration settings, see Chapter 13.16.3.1, "MDIO trigger settings", on page 747.

13.16.4 MDIO decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The "Decode results" table contains information about all decoded frames.

MDIO Decode Results SB1

	State	Start	Start code	OP code	PHYAD/PRTAD	REGAD/DEVAD	DATA/ADDRESS	Register name
1	Opcode Error	-513.292 µs	[bin] 01	[bin] 11	[hex] 0F	[hex] 1A	[hex] DEAD	KSZ8721SL
2	Ok	-487.275 µs	[bin] 00	[bin] 11	[hex] 0E	[hex] 19	[hex] DEAD	KSZ8721CL
3	Ok	-461.258 µs	[bin] 01	[bin] 01	[hex] 0E	[hex] 15	[hex] A56F	KSZ8721CL
4	Ok	-435.197 µs	[bin] 00	[bin] 01	[hex] 0B	[hex] 15	[hex] A55F	KSZ9031MNX
5	Opcode Error	-409.18 µs	[bin] 01	[bin] 00	[hex] 0C	[hex] 17	[hex] CCAF	KSZ8051MNLU
6	Ok	-383.162 µs	[bin] 00	[bin] 00	[hex] 0E	[hex] 17	[hex] BCAF	KSZ8721CL
7	Ok	-357.145 µs	[bin] 01	[bin] 10	[hex] 1A	[hex] 11	[hex] EEFB	KSZ8721BL
8	Ok	-331.084 µs	[bin] 00	[bin] 10	[hex] 1B	[hex] 12	[hex] EEFB	KSZ8721BT

Table 13-25: Content of the Decode results table

Column	Description
Frame	Number of the acquired frame
State	State of frame, available messages are: • OK • OPcode error • Length error • Incomplete frame • Unsynchronized bits
Start	Start time of the frame
Stop	Stop time of the frame
Start code	Start of frame code, 2 bits • "01" (Clause 22) • "00" (Clause 45)
OP code	Operation code (= Frame type), 2 bits • "00" = Address frame (in Clause 45, only) • "01" = Write frame (in Clause 22 or Clause 45) • "10" = Read frame (in Clause 22) or Post Read frame (in Clause 45) • "11" = Read frame (in Clause 45)
PHYAD/PRTAD	Address field, shown as 2 hex characters (corresponding to 5 binary bits) • PHY address (in Clause 22) • Port address (in Clause 45)
REGAD/DEVAD	Address field, shown as 2 hex characters (corresponding to 5 binary bits) • Register address (in Clause 22) • Device address (in Clause 45)
DATA/ADDRESS	Payload data field (in Clause 22 or Clause 45), or Address field (in Clause 45, only), shown as 4 hex characters or 16 binary bits
Register Name	Displays a translation of the PHYAD/PRTAD address label in textual form

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands to retrieve decode results are described in Chapter 24.17.17.3, "Decode results", on page 2045.

13.16.5 Search on decoded MDIO data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

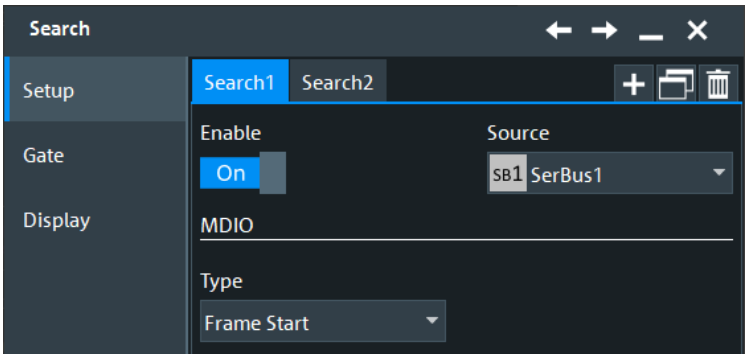
For general information on how to handle the search functionality, see Chapter 11, "Search functions", on page 419.

If you need information on how to get started with searching MDIO data, see Chapter 13.16.5.3, "Searching MDIO data", on page 754. Otherwise proceed with the MDIO search setup.

13.16.5.1 MDIO search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"

Search types
Selects the event types to be searched.



Available event types are "Frame Start", "Frame Stop", and "Data".

Only if search criteria type "Data" is selected, individual search parameters can be specified in the tabs below the "Search criteria" dialog. For these parameters, see "ST" on page 753, "OP" on page 753, "PHYAD/PRTAD" on page 753, "REGAD/DEVAD" on page 753, and "DATA/ADDRESS" on page 753.

Remote command:
`SEARch:TRIGger:MDIO:TYPE` on page 2051

ST
Selects the start of frame code of the frame pattern; available only in search criteria type "Data". Permissible frame patterns are: Clause 22, Clause 45, or Any

Remote command:
`SEARch:TRIGger:MDIO:ST` on page 2050

OP
Selects the Type of Frame code (or OP code, OpCode, operation code); available only in search criteria type "Data". Available frame types are: Address, Write, Read, Post Read, or Any

Remote command:
`SEARch:TRIGger:MDIO:FRAMetype` on page 2049

PHYAD/PRTAD
Sets the physical address or port address (5 bits) of the frame pattern; available only in search criteria type "Data".

Remote command:
`SEARch:TRIGger:MDIO:PHYS` on page 2050

REGAD/DEVAD
Sets the register address or device address (5 bits) of the frame pattern; available only in search criteria type "Data".

Remote command:
`SEARch:TRIGger:MDIO:REGI` on page 2050

DATA/ADDRESS
Defines the payload data pattern or address pattern (16 bits); available only in search criteria type "Data".

Remote command:
`SEARch:TRIGger:MDIO:DATA` on page 2049

13.16.5.2 MDIO search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands:

- `SEARch:RESult:MDIO:WCOunt?` on page 2055
- `SEARch:RESult:MDIO:WORD<m>:STATe?` on page 2053
- `SEARch:RESult:MDIO:WORD<m>:STARt?` on page 2053

- `SEARch:RESult:MDIO:WORD<m>:STOP?` on page 2054
- `SEARch:RESult:MDIO:WORD<m>:ST?` on page 2052
- `SEARch:RESult:MDIO:WORD<m>:TYPE?` on page 2054
- `SEARch:RESult:MDIO:WORD<m>:PHYS?` on page 2052
- `SEARch:RESult:MDIO:WORD<m>:REGI?` on page 2052
- `SEARch:RESult:MDIO:WORD<m>:DATA?` on page 2051
- `SEARch:RESult:MDIO:WORD<m>:SYMBol?` on page 2054

13.16.5.3 Searching MDIO data

Prerequisite: A serial bus is configured for the MDIO signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Tap "Setup search".
3. If the dialog box does not contain a search entry, tap the "Add" icon to create one, as described in "To create a user-defined search" on page 434.
4. Tap "Source" and select the serial bus that is set to MDIO (e.g. "SB1", unless already selected).

   The search dialog for MDIO protocol analysis is opened.
   There are no additional search criteria to be specified.
5. To acquire a waveform, press [Single].

   The R&S RTO6 performs an MDIO decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).
6. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog.

   The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.17 RFFE (option R&S RTO6-K550)

Radio Frequency Front-End (RFFE) control interface is a serial interface specified by the Mobile Industry Processor Interface (MIPI) alliance. The RFFE interface is designed to control RF front-end components in mobile terminals.

13.17.1 The RFFE protocol

The RFFE interface is specified in the "MIPI® Alliance Specification for RF Front-End Control Interface". The RFFE interface is used by the radio frequency front-end interface chips in most LTE-Advanced platforms and in smart phones in general. RFFE is a replacement for existing standards like SPI and I²C that do no meet performance requirements.

Bus structure

RFFE is a two-wire, serial interface that connects up to 4 master devices (Radio Frequency IC, RFIC) to up to 15 slaves (front-end modules, FEM) on a single RFFE bus. A slave device has read-write capability, or it is write only. Only one of the masters is the active master (bus owner master, BOM), which can initiate command sequences on the bus.

The interface has two lines: one clock signal (SCLK) controlled by the master, and a serial bidirectional data signal (SDATA). Furthermore, a VIO supply/reference voltage from a common source is applied to all components on the bus.

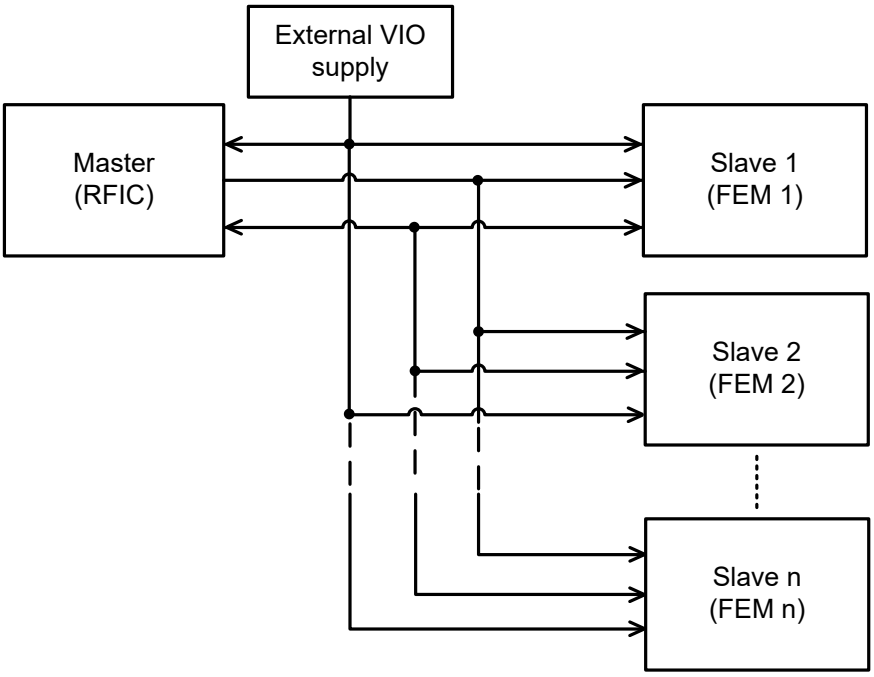


Figure 13-31: RFFE bus structure with external VIO supply

Command sequences

Protocol messages are called command sequences in RFFE. The standard defines various command sequences to accomplish read and write access to slaves and to non-active masters. Command sequences are initiated by the BOM master on the SDATA line.

In general, a command sequence consists of:

- Sequence start condition (SSC)

Two bits: 1 followed by 0 on SDATA while SCLK is at logic level zero.

- Command frame
  Consist of a 4-bit slave address field (SA), followed by 8 command payload bits and a single parity bit.
- Address and/or data frames, depending on the command sequence
  A frame consists of 8 data bits or 8 register address bits, followed by a single parity bit. The number of address and data frames varies depending on the command sequence type.
- Bus park cycle (BP)
  A BP cycle is sent at the end of a command sequence, and when the device transfers control of SDATA to another device.

Between the end of a command sequence and the beginning of a new command sequence, the bus is in idle condition at least for 10 ns.

The bits are sent MSB first.

SSC	Command frame		Address and data frames		BP
	SA 4 bits	Command payload 8 bits + P	[Address frames] 8 bits + P	[Data frames] 8 bits + P	

Figure 13-32: General structure of a RFFE Write command sequence

SSC	Command frame		Address and data frames			BP
	SA 4 bits	Command payload 8 bits + P	[Address frames] 8 bits + P	BP	[Data frames] 8 bits + P	

Figure 13-33: General structure of a RFFE Read command sequence

Trigger

The R&S RTO6 uses a hardware-based trigger to trigger on various parts of slave device messages, to trigger at maximum bus speed and on frame gaps. The data and clock lines must be connected to the input channels. Triggering on math and reference waveforms is not possible.

You can trigger on:

- Start of command sequence (SSC). In addition, you can specify a slave address.
- End of command sequence). In addition, you can specify a slave address.
- Various errors, for example, parity and bus park error
- Read and write command sequences between the BOM and the slaves.
  Within a command sequence, you can trigger on specific parts of the message:
  – Slave address
  – Byte count
  – Register address

– Data word

Search

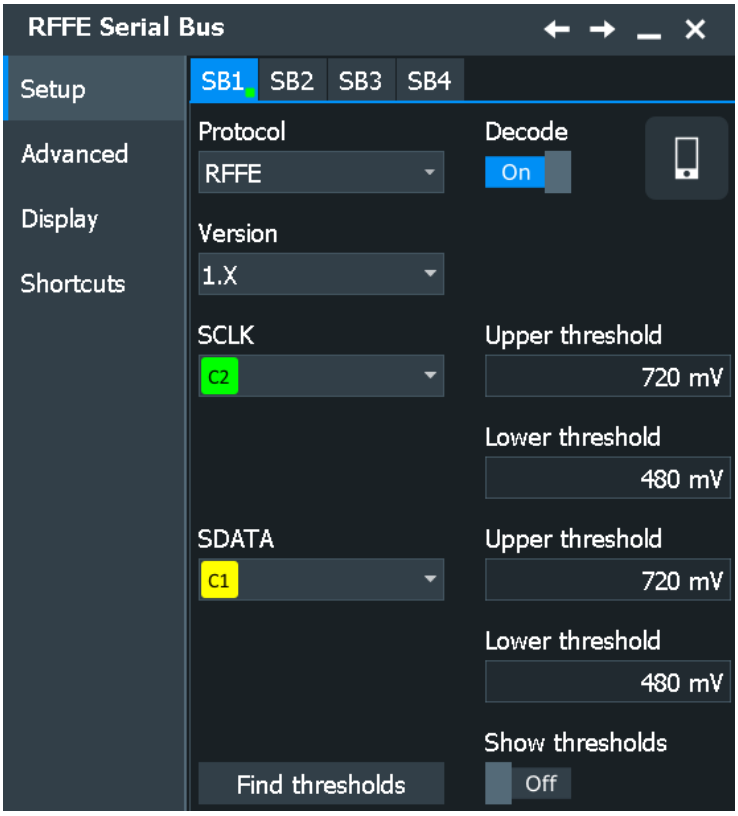
Using the search functionality, you can find various events in the acquired and decoded data. You can find the same events which you also can trigger on. In addition, you can find command sequences of master-to master communication and "interrupt summary and identification" command sequences.

13.17.2 RFFE configuration

13.17.2.1 RFFE configuration settings

Access: [Bus] > "RFFE" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Version
Selects the RFFE version. Supported are 1.x, 2.0, and 2.1.

Remote command:
BUS<m>:RFFE:VERSion on page 2060

Read mode
Selects, if the standard or synchronous read ("sRead") mode is used.

Remote command:
BUS<m>:RFFE:RDMD on page 2059

SCLK Source
Sets the source of the clock line.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Alternatively, digital channels can be used if MSO option R&S RTO6-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Remote command:
BUS<m>:RFFE:CLOCk:SOURce on page 2055

SDATA Source
Sets the source of the data line.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Alternatively, digital channels can be used if MSO option R&S RTO6-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Remote command:
BUS<m>:RFFE:DATA:SOURce on page 2056

Thresholds
Set the threshold values for digitization of signals for each line: the positive going threshold (V_{TP}, upper value), the negative going threshold (V_{TN}, lower value), and the hysteresis. The hysteresis is the difference of V_{TP} and V_{TN}.

If the signal value on the line is higher than the positive going threshold, the signal state is high. Otherwise, the signal state is considered low if the signal value is below the negative going threshold.

There are several ways to set the thresholds

"Threshold"	Enter the value directly in the field.
"Find thresholds"	Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
BUS<m>:RFFE:CLOCk:THReshold:HIGH on page 2057
BUS<m>:RFFE:CLOCk:THReshold:LOW on page 2058
BUS<m>:RFFE:CLOCk:THReshold:HYSTeresis on page 2058

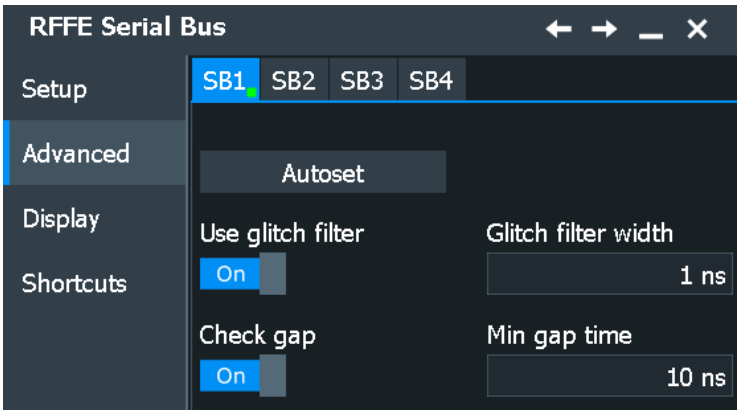
BUS<m>:RFFE:DATA:THReshold:HIGH on page 2056
BUS<m>:RFFE:DATA:THReshold:LOW on page 2057
BUS<m>:RFFE:DATA:THReshold:HYSTeresis on page 2057
BUS<m>:SETReflevels on page 1671
BUS<m>:FAUToset on page 1671

Show thresholds
If enabled, the threshold levels are displayed in the diagram.

Remote command:
BUS<m>:THReshold on page 1671

13.17.2.2 Advanced settings

Access: [Bus] > "Advanced" tab.



Autoset
Starts software algorithms for determining the signal threshold levels and bitrate.

Remote command:
BUS<m>:FAUToset on page 1671

Glitch filter
Enables the glitch filter on the SCLK and SDATA lines to improve decode accuracy.

The "Glitch filter width" field sets the maximum glitch width to be ignored.

Remote command:
BUS<m>:RFFE:GFILter on page 2058
BUS<m>:RFFE:GFWidth on page 2059

Timing
The gap defines the idle time between the Bus Park Cycle (BP) and Sequence Start Condition (SSC).

You can define a "Min gap time". The standard defines a minimum of 10 ns to separate two subsequent command sequences.

Remote command:
`BUS<m>:RFFE:MINGap:SELect` on page 2059
`BUS<m>:RFFE:MINGap:TIME` on page 2059

13.17.2.3 Configuring RFFE signals

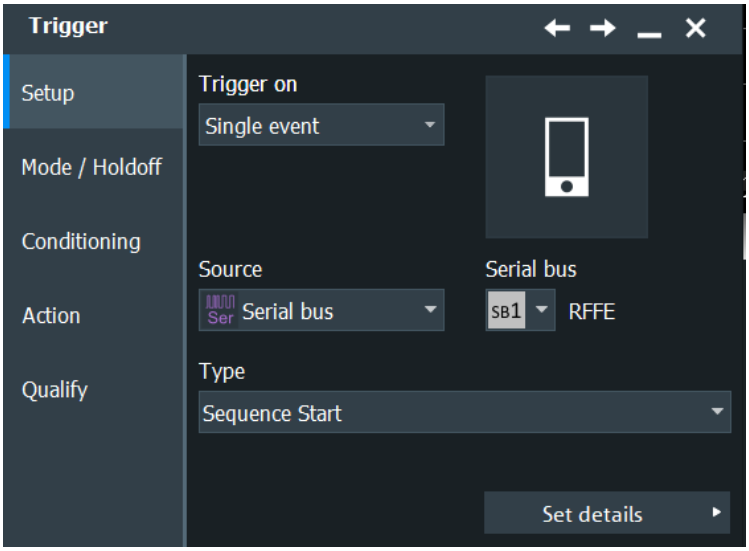
For details on configuration settings, see Chapter 13.17.2.1, "RFFE configuration settings", on page 757.

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. In the "Setup" tab, select the protocol: "RFFE".
4. Select the RFFE "Version".
5. Select the "Read mode".
6. Tap "SCLK", and select the waveform of the clock line.
7. Tap "SDATA", and select the waveform of the data line.
8. Set the logical thresholds.
9. If the signal has glitches which can distort the decoding, enable the glitch filter. Set the glitch width.
10. Enable "Decode".

13.17.3 RFFE trigger

13.17.3.1 RFFE trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Version
Selects the RFFE version. Supported are 1.x, 2.0, and 2.1.

Remote command:
`BUS<m>:RFFE:VERSion` on page 2060

Type
Selects the trigger type for RFFE analysis.

The instrument triggers always at the end of the met trigger criteria.

Remote command:
`TRIGger<m>:RFFE:TYPE` on page 2061

Sequence Start ← Type
Triggers on the beginning of a command sequence, after the slave address. The SSC is a 1 followed by 0 on SDATA line while SCLK is at logic level zero.

Optionally, you can specify a slave address pattern or a slave address range to trigger only on command sequences that are sent to these slaves.

See: "Slave Addr. /MID " on page 766.

Sequence Stop ← Type
Triggers on the end of a command sequence, on the bus park cycle.

Optionally, you can specify a slave address pattern or a slave address range to trigger only on command sequences that are sent to these slaves.

See: "Slave Addr. /MID " on page 766.

Register 0 Write ← Type
Triggers on "Register 0 Write" command sequences. This sequence sends 7 data bits in the command payload.

You can refine the trigger condition:
- Set a slave address condition to trigger only on command sequences that are sent to the specified slaves.
  See: "Slave Addr. /MID " on page 766.
- Set a data pattern condition to trigger on data patterns expected in the message.
  See: "Data" on page 767.

Register Write ← Type
Triggers on "Register Write" command sequences. This sequence sends the register address in the command payload, and sends one data frame.

You can refine the trigger condition:
- Set a slave address condition to trigger only on command sequences that are sent to the specified slaves.
  See: "Slave Addr. /MID " on page 766.
- Set a register address condition to trigger only on command sequences that write data to the specified register.
  See: "Address" on page 766.
- Set a data pattern condition to trigger on data patterns expected in the message.
  See: "Data" on page 767.

Register Read ← Type
Triggers on "Register Read" command sequences. This sequence sends the register address in the command payload, and reads back one data frame.

You can refine the trigger condition using the same settings as for the "Register Write" command sequence, see "Register Write" on page 762.

Extended Register Write ← Type
Triggers on "Extended Register Write" command sequences. This sequence sends the byte count of data frames in the command payload, followed by one address frame with the address of the first extended register, and up to 16 data frames.

You can refine the trigger condition:
- Set a slave address condition to trigger only on command sequences that are sent to the specified slaves.
  See: "Slave Addr. /MID " on page 766
- Set a byte count condition to trigger only on command sequences that send the specified number of data frames.
  See: "Byte count" on page 766
- Set a register address condition to trigger only on command sequences that write data to the specified register.

See: "Address" on page 766
- Set a data pattern condition to trigger on data patterns expected in the message.
  See: "Data" on page 767 and "Index" on page 767

Extended Register Read ← Type
Triggers on "Extended Register Read" command sequences. This sequence sends the byte count of data frames in the command payload, followed by one address frame with the address of the first extended register, and reads back up to 16 data frames.

You can refine the trigger condition using the same settings as for the "Extended Register Write" command sequence, see "Extended Register Write" on page 762.

Extended Register Write Long ← Type
Triggers on "Extended Register Write" command sequences. This sequence sends the byte count of data frames in the command payload, followed by 2 address frames with the address of the first extended register, and up to 8 data frames.

You can refine the trigger condition using the same settings as for the "Extended Register Write" command sequence, see "Extended Register Write" on page 762.

Extended Register Read Long ← Type
Triggers on "Extended Register Write" command sequences. This sequence sends the byte count of data frames in the command payload, followed by 2 address frames with the address of the first extended register, and reads back up to 8 data frames.

You can refine the trigger condition using the same settings as for the "Extended Register Write" command sequence, see "Extended Register Write" on page 762.

Interrupt Summary and Notification ← Type
Available only for "Version"= "2.0"/"2.1".

Triggers on "Interrupt Summary and Notification" command sequence.

You can refine the trigger condition:
- Set a slave address condition to trigger only on command sequences that are sent to the specified MIDs.
  See: "Slave Addr. /MID " on page 766
- Set a bit count condition to trigger only on command sequences that send the specified number of data frames.
  See: "Byte count" on page 766
- Set a data pattern condition to trigger on data patterns expected in the message.
  See: "Data" on page 767

Masked Write ← Type
Available only for "Version"= "2.1".

Triggers on "Masked Write" command sequences. This sequence sends the register address in the command payload, and sends one mask and one data frame.

You can refine the trigger condition:
- Set a slave address condition to trigger only on command sequences that are sent to the specified MIDs.
  See: "Slave Addr. /MID " on page 766.
- Set a register address condition to trigger only on command sequences that write data to the specified register.

See: "Address" on page 766.
- Set a mask pattern condition to trigger on a mask expected in the message.
  See: "Mask" on page 767.
- Set a data pattern condition to trigger on data patterns expected in the message.
  See: "Data" on page 767.

Master Ownership Handover ← Type
Available only for "Version"= "2.0"/"2.1".

Triggers on "Master Ownership Handover" command sequences.

Optionally, you can specify an MID pattern or an MID range to trigger only on command sequences that are sent to these slaves.

Master Write ← Type
Available only for "Version"= "2.0"/"2.1".

Triggers on "Master Write" command sequences. This sequence sends the register address in the command payload, and sends two data frames.

You can refine the trigger condition:
- Set a MID condition to trigger only on command sequences that are sent to the specified MIDs.
  See: "Slave Addr. /MID " on page 766.
- Set a register address condition to trigger only on command sequences that write data to the specified register.
  See: "Address" on page 766.
- Set a data pattern condition to trigger on data patterns expected in the message.
  See: "Data" on page 767.

Master Read ← Type
Available only for "Version"= "2.0"/"2.1".

Triggers on "Master Read" command sequences. This sequence sends the register address in the command payload, and reads two data frames.

You can refine the trigger condition:
- Set a MID condition to trigger only on command sequences that are sent to the specified MIDs.
  See: "Slave Addr. /MID " on page 766.
- Set a register address condition to trigger only on command sequences that write data to the specified register.
  See: "Address" on page 766.
- Set a data pattern condition to trigger on data patterns expected in the message.
  See: "Data" on page 767.

Master Context Transfer Write ← Type
Available only for "Version"= "2.1".

Triggers on "Master Context Transfer Write" command sequences.

You can refine the trigger condition:
- Set an MID condition to trigger only on command sequences that are sent to the specified MIDs.
  See: "Slave Addr. /MID " on page 766

- Set a byte count condition to trigger only on command sequences that send the specified number of data frames.
  See: "Byte count" on page 766
- Set a register address condition to trigger only on command sequences that write data to the specified register.
  See: "Address" on page 766
- Set a data pattern condition to trigger on data patterns expected in the message.
  See: "Data" on page 767 and "Index" on page 767

Master Context Transfer Read ← Type
Available only for "Version"= "2.1".

Triggers on "Master Context Transfer Read" command sequences.

You can refine the trigger condition:
- Set an MID condition to trigger only on command sequences that are sent to the specified MIDs.
  See: "Slave Addr. /MID " on page 766
- Set a byte count condition to trigger only on command sequences that send the specified number of data frames.
  See: "Byte count" on page 766
- Set a register address condition to trigger only on command sequences that write data to the specified register.
  See: "Address" on page 766
- Set a data pattern condition to trigger on data patterns expected in the message.
  See: "Data" on page 767 and "Index" on page 767

Error conditions ← Type
Triggers if at least one of the selected errors occurs in a frame. The instrument triggers on the first detected error.

"SSC error" Triggers when no valid SSC sequence has been found after the idle time. The instrument triggers at the invalid sequence.

"Length error" Triggers on an incorrect length of the command sequence - when additional or missing bits are detected and the sequence of bits is not as expected.

"No response" Triggers on any No Response Frame. All 9 bits of a No Response Frame, including the parity bit, are zero.

"Bus park error"
Triggers on an incorrect bus park cycle - when the instrument expects a low bit in bus park but detects high bit.

"Parity error" Triggers on incorrect parity bits.

"Unknown sequence"
Triggers when the instrument cannot detect any supported command sequence.

"Gap error" Triggers when timing gap error occurs.

"Version error" Triggers on any version errors.

Remote command:
`TRIGger<m>:RFFE:ERRor:SSC` on page 2068
`TRIGger<m>:RFFE:ERRor:LENGth` on page 2068
`TRIGger<m>:RFFE:ERRor:BP` on page 2068
`TRIGger<m>:RFFE:ERRor:PARity` on page 2068
`TRIGger<m>:RFFE:ERRor:NOResponse` on page 2068
`TRIGger<m>:RFFE:ERRor:USEQuence` on page 2069
`TRIGger<m>:RFFE:ERRor:GAP` on page 2069
`TRIGger<m>:RFFE:ERRor:VERSion` on page 2069

Slave Addr. /MID
Defines the address of the slave / master ID. The slave address setup consists of the condition and one or two address patterns.

"Condition"	Sets the operator to trigger on a specific address pattern ("Equal" or "Not equal") or an address range.
"Slave Address (Min)"	Defines the slave address pattern for all operators that require one pattern.
"Slave Adress (Max)"	Defines the second address pattern that is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:RFFE:SADD:CONDition` on page 2066
`TRIGger<m>:RFFE:SADD:MIN` on page 2067
`TRIGger<m>:RFFE:SADD:MAX` on page 2067

Address
Defines the register address. The register address setup consists of the condition and one or two address patterns.

"Condition"	Sets the operator to trigger on a specific address pattern ("Equal" or "Not equal") or an address range.
"Address (Min)"	Defines the register address pattern for all operators that require one pattern.
"Adress (Max)"	Defines the second address pattern that is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:RFFE:ADDRess:CONDition` on page 2063
`TRIGger<m>:RFFE:ADDRess:MIN` on page 2063
`TRIGger<m>:RFFE:ADDRess:MAX` on page 2064

Byte count
Sets the number of data frames to be read or written in the command sequence. The setting is available for all "Extended Register" command sequences, which can transfer more than one data frame.

"Condition"	Sets the operator to trigger on a specific byte count ("Equal" or "Not equal") or a byte count range.
"Byte/Bit count (Min)"	Defines the byte count for all operators that require one count setting.

"Byte/Bit count (Max)" Defines the second byte count value that is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:RFFE:BCOunt:CONDition` on page 2064
`TRIGger<m>:RFFE:BCOunt:MIN` on page 2064
`TRIGger<m>:RFFE:BCOunt:MAX` on page 2064

Data
Defines the data trigger condition. The data setup consists of the condition and one or two data patterns.

To define on which data frames of the sequence you want to trigger, use the Index settings.

"Condition" Sets the operator to trigger on a specific data pattern ("Equal" or "Not equal") or a data range.

"Data (Min)" Defines the data pattern for all operators that require one pattern.

"Data (Max)" Defines the second data pattern that is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:RFFE:DATA:DCON` on page 2065
`TRIGger<m>:RFFE:DATA:DMIN` on page 2065
`TRIGger<m>:RFFE:DATA:DMAX` on page 2065

Index
Defines which data frames are relevant for the trigger. The setting is available for all "Extended Register" command sequences, which can transfer more than one data frame.

The index allows you to check for a certain bit pattern at a certain position in the byte sequence. Furthermore, you can trigger on the occurrence of a certain bit pattern within a data range.

To trigger on any data byte that fulfills the data pattern settings, set the index to XX.

"Condition" Sets the operator to trigger on a specific data frame ("Equal") or a data frame range.

"Data (Min)" Defines the frame index of the data pattern for all operators that require one index.

"Data (Max)" Defines the frame index of the last data pattern that is required to specify a range with condition "In range". The instrument adjusts this value, if you enter a data pattern that is longer than the index range.

Remote command:
`TRIGger<m>:RFFE:DATA:ICONdition` on page 2065
`TRIGger<m>:RFFE:DATA:IMIN` on page 2066
`TRIGger<m>:RFFE:DATA:IMAX` on page 2066

Mask
Defines the mask trigger condition. The mask setup consists of the condition and one or two data patterns.The setting is available for "Masked Write" command sequences.

"Condition" Sets the operator to trigger on a specific mask frame ("Equal") or a mask frame range.

"Mask (Min)" Defines the mask pattern for all operators that require one pattern.

"Mask (Max)" Defines the second mask pattern that is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:RFFE:MASK:CONDition` on page 2067
`TRIGger<m>:RFFE:MASK:MIN` on page 2067
`TRIGger<m>:RFFE:MASK:MAX` on page 2067

13.17.3.2 Triggering on RFFE signals

Prerequisites: An RFFE bus is configured, see Chapter 13.17.2, "RFFE configuration", on page 757, and "Decode" is enabled.

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Tap "Serial bus" and select the serial bus that is set to RFFE.
5. Select the "Trigger type".
6. To refine the trigger settings, configure additional settings, which are available for many trigger types.
   For details, see Chapter 13.17.3.1, "RFFE trigger settings", on page 760.

13.17.4 RFFE label list

Label lists are protocol-specific. A label list file for RFFE contains slave addresses and/or register addresses and their symbolic names:

- "SA": slave address (SID in the labal list file)
- Address: register address.
- "Symbolic label": symbolic name as combination of the slave name (SID text) and the address name (Address text), specifying the device function

Example: RFFE label list file

```
# Labels for RFFE protocol
#   Column order: SID, SID text, Address, Address text
# ---------------------------------------------------------------------------
@PROTOCOL_NAME = rffe
0x01,LM8335
0x01,LM8335,0x00,CMTL_REG
0x01,LM8335,0x01,GPO_PULL_DIR
0x01,LM8335,0x02,GPO_PULL_ENABLE
0x01,LM8335,0x03,GPO_OUT_HIGH_CFG
0x01,LM8335,0x04,GPO_OUT_MASK
0x01,LM8335,0x05,GPO_OUT_DATA
0x01,LM8335,0x1C,PM_TRIG
0x01,LM8335,0x1D,PROD_ID
0x01,LM8335,0x1E,MAN_ID
0x01,LM8335,0x1F,USID_REG
0x05,LM3279
0x05,LM3279,0x00,VSET_CTRL
0x05,LM3279,0x01,STATE_CTRL
0x05,LM3279,0x02,GPO_CTRL
0x05,LM3279,0x1C,PM_TRIG
0x05,LM3279,0x1D,PROD_ID
0x05,LM3279,0x1E,MAN_ID
0x05,LM3279,0x1F,USID_REG
0x0A,Test
```

See also: Chapter 13.1.4, "Content and format of the PTT file", on page 489.

Remote command:

- `BUS<m>:RFFE:SEQuence<n>:SYMBol?` on page 2073

13.17.5 RFFE decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

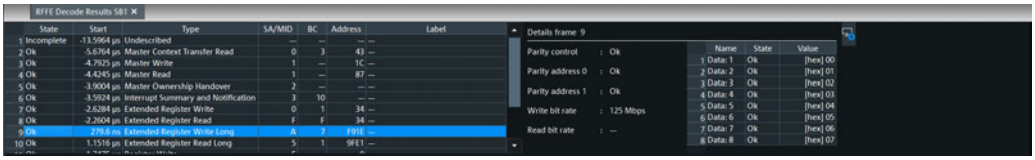


Table 13-26: Content of the Decode results table

Column	Description
State	Overall state of the frame. "Insuffcient waveform length" indicates that the frame is not completely contained in the acquisition. Change the horizontal scale, or move the reference point to the left to get a longer acquisition.
Start	Time of command sequence start (SSC)
Type	Type of the command sequence
SA /MID	Address of the SA/ MID
BC	Byte count (decimal value)
Address	Register address (hex value)
Label	Symbolic name of the address if a label list is applied.

Table 13-27: Content of the Details frame table

Column	Description
Parity control	Checks if there an error in the parity
Parity address 0/1	Checks if there an error in the parity address
Read bit rate	Value of the read bit rate
Write bit rate	Value of the write bit rate
Name	Type of the command sequence
State	Overall state of the frame. "Insuffcient waveform length" indicates that the frame is not completely contained in the acquisition. Change the horizontal scale, or move the reference point to the left to get a longer acquisition.
Value	Values of the data bytes.

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands to retrieve decode results are described in Chapter 24.17.18.3, "Decode results", on page 2069.

13.17.6 Search on decoded RFFE data

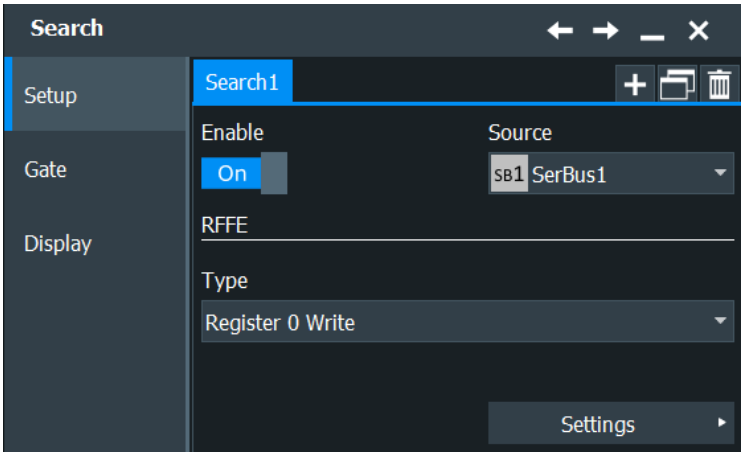
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 11, "Search functions", on page 419.

13.17.6.1 RFFE search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"



Type
The search criterion is defined by "Type". All trigger types are also available for search.

Additional search parameters are provided under "Detailed Search Parameter Setup".

For details, see also "Type" on page 761.

"Sequence Start"
Searches for SSC.
Additional search parameter: "Slave Addr." on page 773

"Sequence Stop"
Searches for bus park at the end of a command sequence.
Additional search parameter: "Slave Addr." on page 773

"Register 0 Write"
Searches for Register 0 Write command sequences. Additional search parameters:
- Slave address, see "Slave Addr." on page 773.
- Data pattern, see "Data" on page 774

"Register Write"
Searches for Register Write command sequences. Additional search parameters:
- Slave address, see "Slave Addr." on page 773
- Register address, see "Address" on page 773
- Data pattern, see "Data" on page 774

"Register Read"
Searches for Register Read command sequences.
This search uses the same additional parameters as the Register Write search.

"Extended Register Write"
Searches for Extended Register Write command sequences. Additional search parameters:
- Slave address, see "Slave Addr." on page 773
- Byte count, see "Byte count" on page 773
- Register address, see "Address" on page 773
- Data pattern, see "Data" on page 774 and "Index" on page 774

"Extended Register Read"
Searches for Extended Register Read command sequences.
This search uses the same additional parameters as the Extended Register Write search.

"Extended Register Write Long"
Searches for Extended Register Write Long command sequences.
This search uses the same additional parameters as the Extended Register Write search.

"Extended Register Read Long"
Searches for Extended Register Read Long command sequences.
This search uses the same additional parameters as the Extended Register Write search.

"Master Read"
Searches for Master Read command sequences. Additional search parameters:
- Address of the non-BOM master, see "Slave Addr." on page 773.
- Register address, see "Address" on page 773.
- Data pattern that is returned by the non-BOM master, see "Data" on page 774.

"Master Write"
Searches for Master Write command sequences. Additional search parameters:
- Address of the non-BOM master, see "Slave Addr." on page 773
- Register address, see "Address" on page 773
- Data pattern to be written into the non-BOM, see "Data" on page 774.

"Master Ownership Handover"
Searches for Master Ownership Handover command sequences. Additional search parameters:
- Address of the BOM-to-be, see "Slave Addr." on page 773
- Data pattern of the confirmation sequence of the new BOM, see "Data" on page 774.

"Interrupt Summary and Notification"
Searches for interrupts. Additional search parameters:
- Slave address, see "Slave Addr." on page 773
- Byte count, see "Byte count" on page 773
- Data of the interrupt identification sequence, see

"Error Condition"
Searches for the selected errors, see "Error conditions" on page 774.

Remote command:
`SEARch:TRIGger:RFFE:TYPE` on page 2076

Slave Addr.
Searches for a slave address or a slave address range.

In command sequences transferred between masters, this is the non-BOM address.

The slave address settings are the same as in the RFFE trigger setup, see "Slave Addr. /MID " on page 766.

Remote command:
`SEARch:TRIGger:RFFE:SADD:CONDition` on page 2078
`SEARch:TRIGger:RFFE:SADD:MIN` on page 2080
`SEARch:TRIGger:RFFE:SADD:MAX` on page 2080

Byte count
Searches for a byte count or a byte count range.

The byte count settings are the same as in the RFFE trigger setup, see "Byte count" on page 766

Remote command:
`SEARch:TRIGger:RFFE:BCOunt:CONDition` on page 2079
`SEARch:TRIGger:RFFE:BCOunt:MIN` on page 2080
`SEARch:TRIGger:RFFE:BCOunt:MAX` on page 2080

Address
Searches for a register address or an address range.

The register address settings are the same as in the RFFE trigger setup, see "Address" on page 766.

Remote command:
`SEARch:TRIGger:RFFE:ADDRess:CONDition` on page 2078
`SEARch:TRIGger:RFFE:ADDRess:MIN` on page 2080
`SEARch:TRIGger:RFFE:ADDRess:MAX` on page 2080

Data
Searches for a data pattern or a data word range. The setting is available for all "Extended Register" command sequences, which can transfer more than one data frame.

The data settings are the same as in the RFFE trigger setup, see "Data" on page 767.

Remote command:
`SEARch:TRIGger:RFFE:DATA:DCON` on page 2079
`SEARch:TRIGger:RFFE:DATA:DMIN` on page 2080
`SEARch:TRIGger:RFFE:DATA:DMAX` on page 2080

Index
Defines which data frames are relevant for the search. The setting is available for all Extended Register command sequences, which can transfer more than one data frame. To search for any data byte that fulfills the data pattern settings, set the index to XX.

Remote command:
`SEARch:TRIGger:RFFE:DATA:ICONdition` on page 2079
`SEARch:TRIGger:RFFE:DATA:IMIN` on page 2080
`SEARch:TRIGger:RFFE:DATA:IMAX` on page 2080

Data
Defines the pattern of the interrupt identification sequence, which consists of interrupt slots 15 to 0. The setting is only available for "Interrupt Summary and Notification" command sequence.

Remote command:
`SEARch:TRIGger:RFFE:INTerrupt` on page 2080

Error conditions
Defines which error conditions are relevant for the search.

Remote command:
`SEARch:TRIGger:RFFE:ERRor:LENGth` on page 2081
`SEARch:TRIGger:RFFE:ERRor:PARity` on page 2081
`SEARch:TRIGger:RFFE:ERRor:SSC` on page 2081
`SEARch:TRIGger:RFFE:ERRor:GAP` on page 2082
`SEARch:TRIGger:RFFE:ERRor:VERSion` on page 2082

13.17.6.2 RFFE search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands:

- `SEARch:RESult:RFFE:FCOunt?` on page 2083
- `SEARch:RESult:RFFE:SEQuence<m>:TYPE?` on page 2083
- `SEARch:RESult:RFFE:SEQuence<m>:STATe?` on page 2083
- `SEARch:RESult:RFFE:SEQuence<m>:STARt?` on page 2083
- `SEARch:RESult:RFFE:SEQuence<m>:STOP?` on page 2084
- `SEARch:RESult:RFFE:SEQuence<m>:SADD?` on page 2084
- `SEARch:RESult:RFFE:SEQuence<m>:ADDRess?` on page 2084
- `SEARch:RESult:RFFE:SEQuence<m>:BCOunt?` on page 2085
- `SEARch:RESult:RFFE:SEQuence<m>:DATA?` on page 2085
- `SEARch:RESult:RFFE:SEQuence<m>:SYMBol?` on page 2085
- `SEARch:RESult:RFFE:SEQuence<m>:BYTE<n>:STATe?` on page 2085
- `SEARch:RESult:RFFE:SEQuence<m>:BYTE<n>:VALue?` on page 2086

13.17.6.3 Searching RFFE data

Prerequisites: An RFFE bus is configured, see Chapter 13.17.2, "RFFE configuration", on page 757, and "Decode" is enabled.

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Tap "Setup search".
3. If the dialog box does not contain a search entry, tap the "Add" icon to create one, as described in "To create a user-defined search" on page 434.
4. Tap the "Source" button and select the serial bus that is set to RFFE.

   "Protocol" shows the RFFE icon.
5. Tap "Type" and select the search type.
   All trigger types are also available for search.
6. To refine the search settings, configure additional settings, which are available for many search types.
   For details, see Chapter 13.17.6.1, "RFFE search setup", on page 771.
7. Under "Control", "Enable" the search.

   The "Search Results" box opens.
8. Close the "Search" dialog box.
9. Press [Run Stop] to start acquisition.
10. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

    The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.18 Ethernet 100BASE-T1 (option R&S RTO6-K560)

100BASE-T1 is an Ethernet based protocol specialized for the needs of the automotive industry, such as in the automotive networking applications, enabling advanced safety, comfort and infotainment features within the automobile.

Its specifications, with reference to IEEE Standard 802.3-2012, were developed by the OPEN (One-Pair Ether-Net) alliance under BroadR-Reach®.

BroadR-Reach is a Broadcom® point-to-point Ethernet physical layer (PHY) technology. Therefore, 100BASE-T1 is also known as BroadR-Reach Physical Layer (BR-PHY) or Open Alliance BroadR-Reach PHY (OABR PHY). For more details on the specifications, refer to http://www.opensig.org/about/specifications/.

This option is compatible with the MII (IEEE 802.3 Clause 22) and IEEE 802.3 MAC operating at 100Mbps.

- 100BASE-T1 basics....776
- 100BASE-T1 configuration....777
- 100BASE-T1 trigger....782
- 100BASE-T1 decode results....787
- Search on decoded 100BASE-T1 data....789

13.18.1 100BASE-T1 basics

The BR-PHY has the following objectives:

- Provides a PHY that supports full duplex operating at 100 Mbps over a pair of unshielded twisted pair (UTP) cable or better cable
- Provides compatibility with the Media Independent Interface (MII) IEEE 802.3 Clause 22 and IEEE 802.3 Media Access Controller (MAC) operating at 100 Mbps
- Achieves bit error rate (BER) of less than $1e^{-10}$

To achieve the above objectives, 100BASE-T1 uses 1000BASE-T PHYs with parts of IEEE 802.3 100BASE-TX in operation at 100 MBps and develops a new solution for the PHY sublayers, i.e. Physical Coding Sublayer (PCS) and Physical Medium Attachment (PMA) sublayer as follows:

- Adopts full duplex communication of 1000BASE-T and therefore echo cancellation on a single twisted-pair channel. This feature reduces cabling while preserving the Ethernet MAC compatibility
- Adopts Pulse Amplitude Modulation (PAM-3) encoding scheme with the following encoding techniques:
  - Data encoding is carried out via a 4b3b encoder that converts the MII data (4B - four bits) with 25 MHz clock to three bits (3B) wide of data that is transmitted during one 33.3 MHz clock period

– Symbol encoding is carried out via a one-dimensional (1D) PAM-3 encoder that converts the 3-bit groups into pairs of ternary symbols. These symbols are transmitted using three voltage levels (-1 V, 0 V and +1 V). One symbol is transmitted in each symbol period
– Data scrambling is carried out via a sidestream scramble to randomize the sequence of transmitted symbols and avoid the presence of spectral lines in the signal spectrum

PAM-3 has a higher spectral efficiency that limits the signaling bandwidth to 33.3 MHz instead of 65 MHz to 80 MHz in 1000BASE-T and 100BASE-TX such that communication occurs in the best part of a twisted-pair channel. This in turn improves return loss, reduces crosstalk and EMI, allows for a more aggressive Electromagnetic Compatibility (EMC) filtering and lower cost (often lower quality) cabling

In terms of the trigger and decoder development, multiple aspects of this protocol are new, specifically:

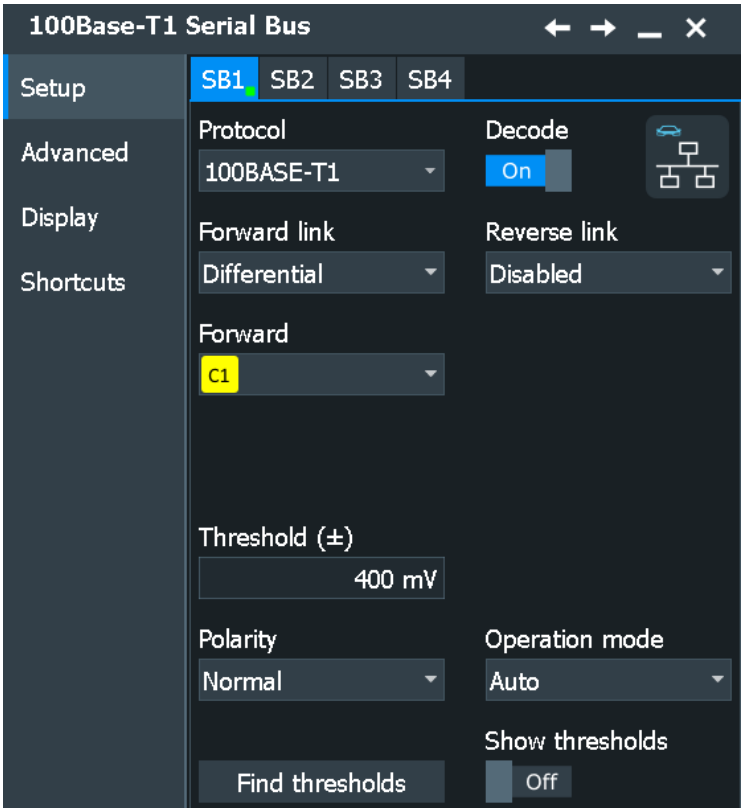
- Sidestream descrambler: Descrambles the randomized sequence of received ternary symbols
- 1D PAM-3 decoder: Decodes the received ternary symbols into groups of three bits according to a conversion chart
- 4b/3b line decoder: Decodes the groups of three bits (result of ternary pair) back to groups of four bits of the data stream

13.18.2 100BASE-T1 configuration

13.18.2.1 100BASE-T1 configuration settings

Access: [Bus] > "100BASE-T1" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Forward

Defines the forward signal settings.

You can choose between a differential and single-ended signal and set the forward source channel for the waveform. Only analog channels are available for selection.

Remote command:

`BUS<m>:HBTO:FTYP` on page 2088
`BUS<m>:HBTO:FDIF` on page 2087
`BUS<m>:HBTO:FDAP` on page 2087
`BUS<m>:HBTO:FDAM` on page 2087

Reverse

Defines the reversed signal settings. The reverse direction is optional and can be disabled.

You can choose between a differential and single-ended signal and set the reverse source channel for the waveform. Only analog channels are available for selection.

Remote command:

`BUS<m>:HBTO:RTYP` on page 2089
`BUS<m>:HBTO:RDIF` on page 2088
`BUS<m>:HBTO:RDAP` on page 2088
`BUS<m>:HBTO:RDAM` on page 2088

Reverse attenuation
Sets the attenuation factor. It is used to de-amplify the reverse signal before subtracting it from the forward signal.

The effective signal amplitude passed to the decoder is:

$$Amp = (Fwd_{+} - Fwd_{-}) - 10^{\frac{Attn}{20}}(Rev_{+} - Rev_{-})$$

Remote command:
`BUS<m>:HBTO:ATTN` on page 2087

Polarity
Selects the polarity of the data signal. You can select between "Normal" and "Inverted".

For "Normal" polarity, active high is defined as +1 V and active low is defined as -1 V.
For "Inverted" polarity, active high is defined as -1 V and active low is defined as +1 V.

Remote command:
`BUS<m>:HBTO:POLarity` on page 2089

Thresholds
Sets the positive and negative thresholds for the digitalization of the signal line.

The signal can have three states:

- High: the signal value is higher than the positive threshold.
- Low: the signal value is below the negative threshold.
- Zero: signal is between the positive and the negative threshold.

"Thresholds"	Enter the value directly.
"Find thresholds"	Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
`BUS<m>:HBTO:THReshold` on page 2089
`BUS<m>:SETReflevels` on page 1671
`BUS<m>:FAUToset` on page 1671

Operation mode
Selects the desired direction of the full-duplex signal for analysis. If set to "Auto", the decoder automatically detects the "Master" or "Slave" mode.

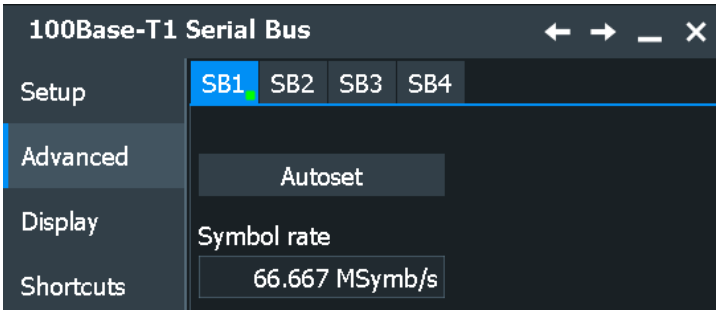
Remote command:
`BUS<m>:HBTO:MODE` on page 2089

Show thresholds
If enabled, the threshold levels are displayed in the diagram.

Remote command:
`BUS<m>:THReshold` on page 1671

13.18.2.2 Advanced settings

Access: [Bus] > "Advanced" tab.



Symbol rate
Defines the transmission rate of ternary symbols which is by default 66.67 MSymb/s. This parameter should be rarely changed.

Remote command:
`BUS<m>:HBTO:SYMRate` on page 2089

Autoset
Starts software algorithms for determining the signal threshold levels and bitrate.

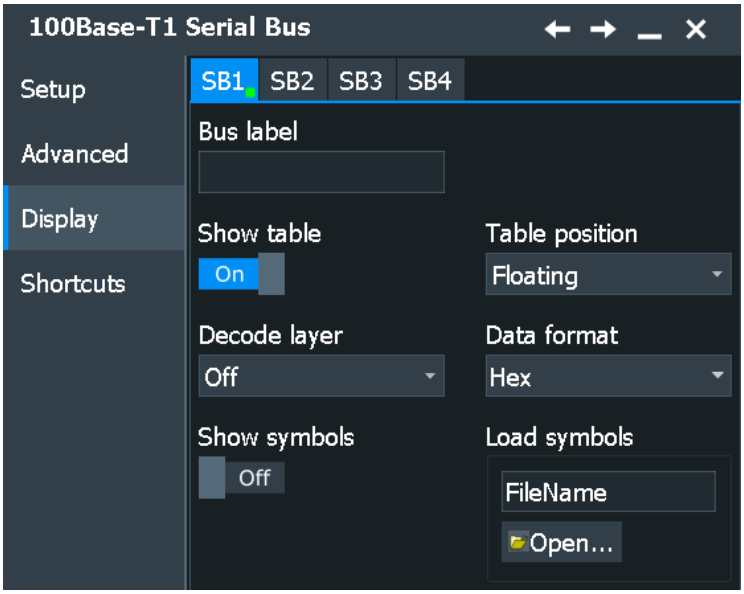
Remote command:
`BUS<m>:FAUToset` on page 1671

13.18.2.3 100BASE-T1 display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

To enhance the decode possibilities of the 100BASE-T1 protocol, you can use an additional setting in the "Display" tab: "Decode layer".



100BASE-T1 Specific Symbol Settings

Label lists are protocol-specific. Label lists for 100BASE-T1 are available in CSV and PTT format.

A 100BASE-T1 label file contains two values for each ethernet header:

- Ethernet Header Display
- Symbolic label: name of the address, specifiing its function in the bus network.

Decode layer
Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

"Off"	No decode layer is displayed.
"Ternary symbols"	Decoded ternary symbols.
"Scrambled bits"	Scrambled ternary bits.
"Descrambled bits"	Descrambled ternary bits.
"Reversed Bits"	Reserved bits

13.18.2.4 Configuring 100BASE-T1 signals

For configuration, assign the lines to the input channels and define the active states and the logical thresholds.

For details on configuration settings, see Chapter 13.18.2.1, "100BASE-T1 configuration settings", on page 777.

Serial bus setup

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. In the "Setup" tab, select the protocol: "100BASE-T1".
4. Select the type and source for the reversed and forward signal.
5. Select the polarity ("Normal" or "Inverted") of the signal.
6. Set the logical thresholds.
7. Enable "Decode".

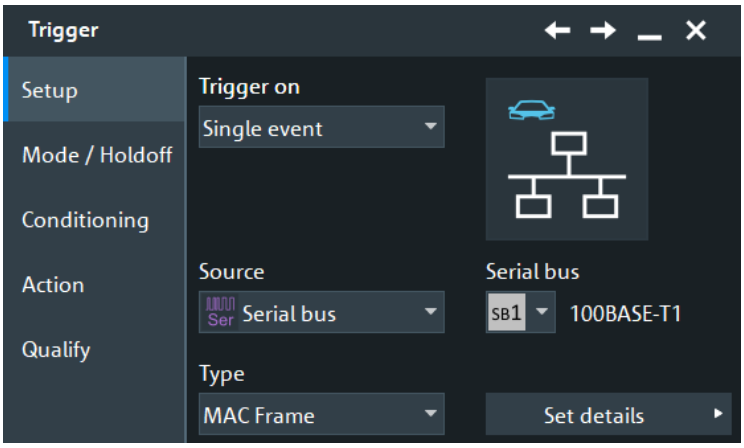
13.18.3 100BASE-T1 trigger

13.18.3.1 100BASE-T1 trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"

In this section, all trigger settings are described. Their availability on the instrument depends on the selected trigger type. The user interface of the instrument displays only appropriate settings and guides you through the trigger setup.

For a list of supported trigger conditions, refer to data sheet.



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Type
Selects the trigger type for 100BASE-T1 analysis.

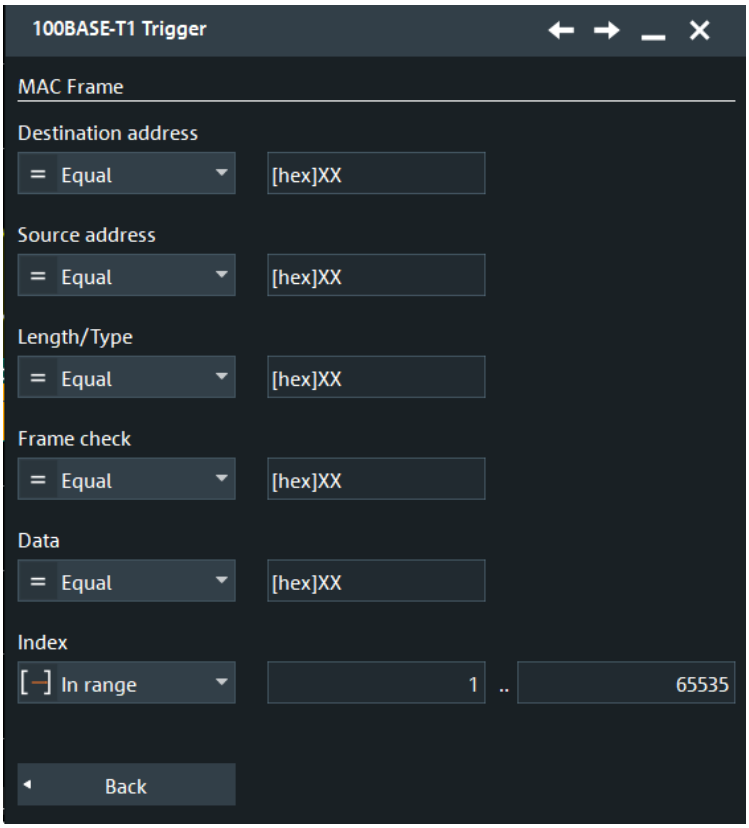
Remote command:
`TRIGger<m>:HBTO:TYPE` on page 2090

Frame Start ← Type
Triggers on the start of frame.

MAC Frame ← Type
Triggers on a Media Access Control (MAC) frame. You can specify:

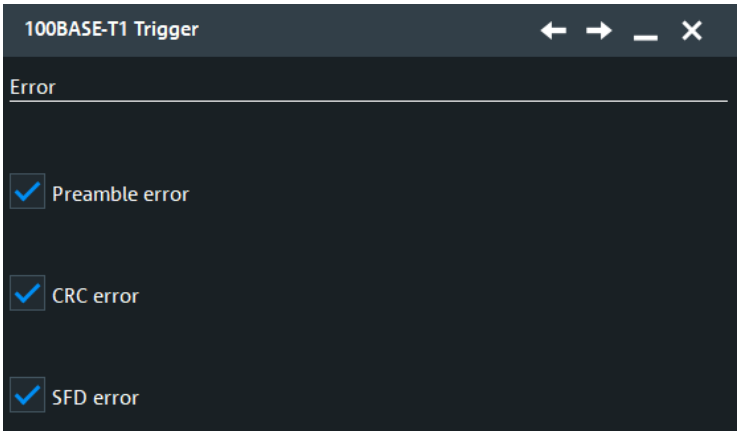
- Destination address
- Source address
- Length/Type
- Frame check
- Data
- Index

The frame contains addresses of the devices and MAC control information that define how to go about transmitting and receiving frames.



IDLE ← Type
Triggers on an idle frame. The frame is used for clock synchronization.

Error ← Type
Triggers on the specified error frame.



Destination address
Sets the specified destination address to be triggered on. The destination address setup consists of the condition and one or two data patterns.

The destination address corresponds to the address of the interface in the device that receives the frame. If the destination address does not match the interface's own Ethernet address, then the interface is free to ignore the rest of the frame.

"Condition" Defines the operator to set a specific destination address, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the destination address pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:HBTO:DADDress:CONDition` on page 2091
`TRIGger<m>:HBTO:DADDress:MIN` on page 2091
`TRIGger<m>:HBTO:DADDress:MAX` on page 2092

Source address
Sets the specified source address to be triggered on. The source address setup consists of the condition and one or two data patterns.

The source address is the physical address of the device that sends the frame.

"Condition" Defines the operator to set a specific source address, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the source address pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:HBTO:SADDress:CONDition` on page 2092
`TRIGger<m>:HBTO:SADDress:MIN` on page 2092
`TRIGger<m>:HBTO:SADDress:MAX` on page 2093

Length/Type
Sets the specified Length/Type to be triggered on. The Length/Type setup consists of the condition and one or two data patterns.

The value in this field indicates the manner in which the field is being used, either as a length or type field. As a length field, the value in the field indicates the number of logical link control (LLC) data octets that follow in the data field of the frame. As a type field, the value in the field is used to indicate the type of protocol data being carried in the data field of the frame.

"Condition" Defines the operator to set a specific Length/Type, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the Length/Type pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:HBTO:LENGth:CONDition` on page 2093
`TRIGger<m>:HBTO:LENGth:MIN` on page 2093
`TRIGger<m>:HBTO:LENGth:MAX` on page 2093

Frame check
Sets the specified frame check to be triggered on. The frame check setup consists of the condition and one or two data patterns.

This field contains a value that is used to check the integrity of the various bits in the frame fields (excluding the preamble / SFD).

"Condition" Defines the operator to set a specific frame check, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the frame check pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:HBTO:CRC:CONDition` on page 2094
`TRIGger<m>:HBTO:CRC:MIN` on page 2094
`TRIGger<m>:HBTO:CRC:MAX` on page 2094

Data
Sets the specified data to be triggered on. The data setup consists of the condition and one or two data patterns.

"Condition" Defines the operator to set a specific data, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the data pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:HBTO:DATA:DCONdition` on page 2095
`TRIGger<m>:HBTO:DATA:DMIN` on page 2095
`TRIGger<m>:HBTO:DATA:DMAX` on page 2095

Index
Sets the specified value or range within this series of data that is considered for the analysis. The index setup consists of the condition and one or two index values.

"Condition" Defines the operator to set a specific index, e.g. "Equal" or a range.

"Index Min/Index"
Defines the bit pattern of the index pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Index Max" The second index pattern is required to specify a range with conditions "In range".

Remote command:
`TRIGger<m>:HBTO:DATA:ICONdition` on page 2096
`TRIGger<m>:HBTO:DATA:IMIN` on page 2096
`TRIGger<m>:HBTO:DATA:IMAX` on page 2096

Errors
Sets the type of error events to be triggered on.

Preamble Error ← Errors
Triggers on a frame with invalid preamble.

Remote command:
`TRIGger<m>:HBTO:ERRor:PREamble` on page 2096

CRC Error ← Errors
Triggers on a frame that has a mismatch of the Cyclic Redundancy Check (CRC) value between the transmitting and receiving device.

Remote command:
`TRIGger<m>:HBTO:ERRor:CRC` on page 2097

SFD Error ← Errors
Triggers on a frame with invalid Start Frame Delimiter (SFD).

Remote command:
`TRIGger<m>:HBTO:ERRor:SFD` on page 2097

13.18.3.2 Triggering on 100BASE-T1

Prerequisite: A bus is configured for the 100BASE-T1 signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to 100BASE-T1.
5. Tap "Type" and select the trigger type to be used for 100BASE-T1 protocol analysis.
6. To refine the trigger settings, configure additional settings, which are available for some trigger types.

For details, see Chapter 13.18.3, "100BASE-T1 trigger", on page 782.

13.18.4 100BASE-T1 decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

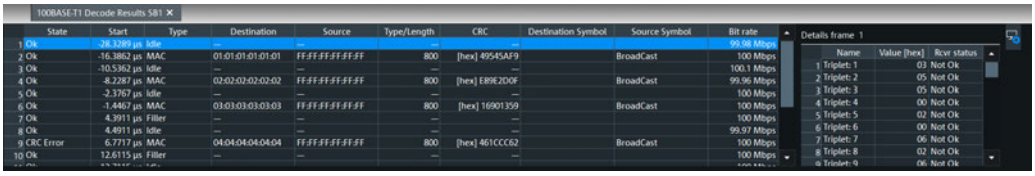


Table 13-28: Content of the Decode results table

Column	Description
Frame	Frame count
State	Overall state of the frame: either OK or the relevant error condition (preamble, CRC or SFD)
Start	Start time of the frame
Type	Type of frame (e.g. Idle, MAC or data)
Destination	Destination address of the frame
Source	Source address of the frame
Type/Length	The sub-protocol (e.g. HTML, video, etc.) determines what meaning this field has. Since the content of this data area is not decoded, the interpretation of this field is ambivalent. It could either be the word type (specific for the sub-protocol) or the word length.
CRC	Frame Check (Cyclic Redundancy Code, CRC)
Destination symbol	Symbol defined for the respective destination
Source symbol	Symbol defined for the respective source
Bit rate	Value of the bit rate

Table 13-29: Content of the Details frame table

Column	Description
Name	State of the field
Type	Field type
Value	Hexadecimal value of the field
Rcvr status	Receiver status

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands are described in Chapter 24.17.19.3, "Decode results", on page 2097.

13.18.5 Search on decoded 100BASE-T1 data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

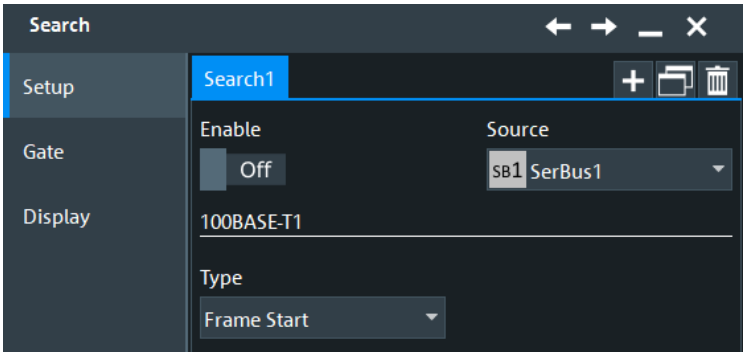
Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 11, "Search functions", on page 419.

13.18.5.1 100BASE-T1 search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"



Enable
Enables the search for 100BASE-T1.

Type
Selects the search type for 100BASE-T1 analysis.

Remote command:
`SEARch:TRIGger:HBTO:TYPE` on page 2103

Frame Start ← Type
Searches for the start of frame.

MAC Frame ← Type
Searches for a Media Access Control (MAC) frame that matches the specified Destination address, Source address, Length/Type, Frame check, Data or Index condition.

The frame contains addresses of the devices and MAC control information that define how to go about transmitting and receiving frames.

FILLER ← Type
Searches for a Filler frame. The frame is used to maintain transmission activity.

IDLE ← Type
Searches for an idle frame. The frame is used for clock synchronization.

Error ← Type
Searches for the specified error frame.

Destination address
Sets the specified destination address to be searched for. The destination address setup consists of the condition and one or two data patterns.

The destination address corresponds to the address of the interface in the device that receives the frame. If the destination address does not match the interface's own Ethernet address, then the interface is free to ignore the rest of the frame.

"Condition" Defines the operator to set a specific destination address, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the destination address pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:HBTO:DADDress:CONDition` on page 2104
`SEARch:TRIGger:HBTO:DADDress:MIN` on page 2104
`SEARch:TRIGger:HBTO:DADDress:MAX` on page 2105

Source address
Sets the specified source address to be searched for. The source address setup consists of the condition and one or two data patterns.

The source address is the physical address of the device that sends the frame.

"Condition" Defines the operator to set a specific source address, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the source address pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
SEARch:TRIGger:HBTO:SADDress:CONDition on page 2105
SEARch:TRIGger:HBTO:SADDress:MIN on page 2105
SEARch:TRIGger:HBTO:SADDress:MAX on page 2106

Length/Type
Sets the specified Length/Type to be searched for. The Length/Type setup consists of the condition and one or two data patterns.

The value in this field indicates the manner in which the field is being used, either as a length or type field. As a length field, the value in the field indicates the number of logical link control (LLC) data octets that follow in the data field of the frame. As a type field, the value in the field is used to indicate the type of protocol data being carried in the data field of the frame.

"Condition" Defines the operator to set a specific Length/Type, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the Length/Type pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
SEARch:TRIGger:HBTO:LENGth:CONDition on page 2106
SEARch:TRIGger:HBTO:LENGth:MIN on page 2106
SEARch:TRIGger:HBTO:LENGth:MAX on page 2107

Frame check
Sets the specified frame check to be searched for. The frame check setup consists of the condition and one or two data patterns.

This field contains a value that is used to check the integrity of the various bits in the frame fields (excluding the preamble / SFD).

"Condition" Defines the operator to set a specific frame check, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the frame check pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
SEARch:TRIGger:HBTO:CRC:CONDition on page 2107
SEARch:TRIGger:HBTO:CRC:MIN on page 2107
SEARch:TRIGger:HBTO:CRC:MAX on page 2108

Data
Sets the specified data to be searched for. The data setup consists of the condition and one or two data patterns.

"Condition" Defines the operator to set a specific data, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the data pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:HBTO:DATA:DCONdition` on page 2108
`SEARch:TRIGger:HBTO:DATA:DMIN` on page 2108
`SEARch:TRIGger:HBTO:DATA:DMAX` on page 2109

Index
Sets the specified value or range within this series of data that is considered for the search. The index setup consists of the condition and one or two index values.

"Condition" Defines the operator to set a specific index, e.g. "Equal" or a range.

"Index Min/Index"
Defines the bit pattern of the index pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Index Max" The second index pattern is required to specify a range with conditions "In range".

Remote command:
`SEARch:TRIGger:HBTO:DATA:ICONdition` on page 2109
`SEARch:TRIGger:HBTO:DATA:IMIN` on page 2109
`SEARch:TRIGger:HBTO:DATA:IMAX` on page 2110

Errors
Sets the type of error events to be searched for.

Preamble Error ← Errors
Searches for a frame with invalid preamble.

Remote command:
`SEARch:TRIGger:HBTO:ERRor:PREamble` on page 2110

CRC Error ← Errors
Searches for a frame that has a mismatch of the Cyclic Redundancy Check (CRC) value between the transmitting and receiving device.

Remote command:
`SEARch:TRIGger:HBTO:ERRor:CRC` on page 2110

SFD Error ← Errors
Searches for a frame with invalid Start Frame Delimiter (SFD).

Remote command:
`SEARch:TRIGger:HBTO:ERRor:SFD` on page 2110

13.18.5.2 100BASE-T1 search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands are described in Chapter 24.17.19.5, "Search results", on page 2111.

13.18.5.3 Searching 100BASE-T1 data

Prerequisite: A serial bus is configured for the 100BASE-T1 signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Tap "Setup search".
3. If the dialog box does not contain a search entry, tap the "Add" icon to create one, as described in "To create a user-defined search" on page 434.
4. Tap "Source" and select the serial bus that is set to 100BASE-T1 (e.g. "SB1", unless already selected).

   The search dialog for 100BASE-T1 protocol analysis opens.
5. Specify search criteria according to Chapter 13.18.5, "Search on decoded 100BASE-T1 data", on page 789.
6. To acquire a waveform, press [Single].

   The R&S RTO6 performs an 100BASE-T1 decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).
7. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog.

   The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.19 Ethernet 1000BASE-T1 (option R&S RTO6-K560)

1000BASE-T1 is a single pair Ethernet based protocol specialized for the needs of the automotive industry, such as in the automotive networking applications, enabling advanced safety, comfort and infotainment features within the automobile. Its specifications is reference to IEEE P802.3bp.

- 1000BASE-T1 basics....794
- 1000BASE-T1 configuration....795
- 1000BASE-T1 trigger....798
- 1000BASE-T1 decode results....804
- Search on decoded 1000BASE-T1 data....806

13.19.1 1000BASE-T1 basics

The 1000BASE-T1 has the following objectives:

- Provides a PHY that supports full duplex operating at 1000 Mbps over a pair of unshielded sheathed twisted-pair cable or better cable.
- Achieves bit error rate (BER) of less than $1e^{-10}$.
- Support for up to at least 15 meters of cable length in the automotive link segment.

To achieve the above objectives, 1000BASE-T1 uses Physical Coding Sublayer (PCS) and Physical Medium Attachment (PMA) sublayer as follows:

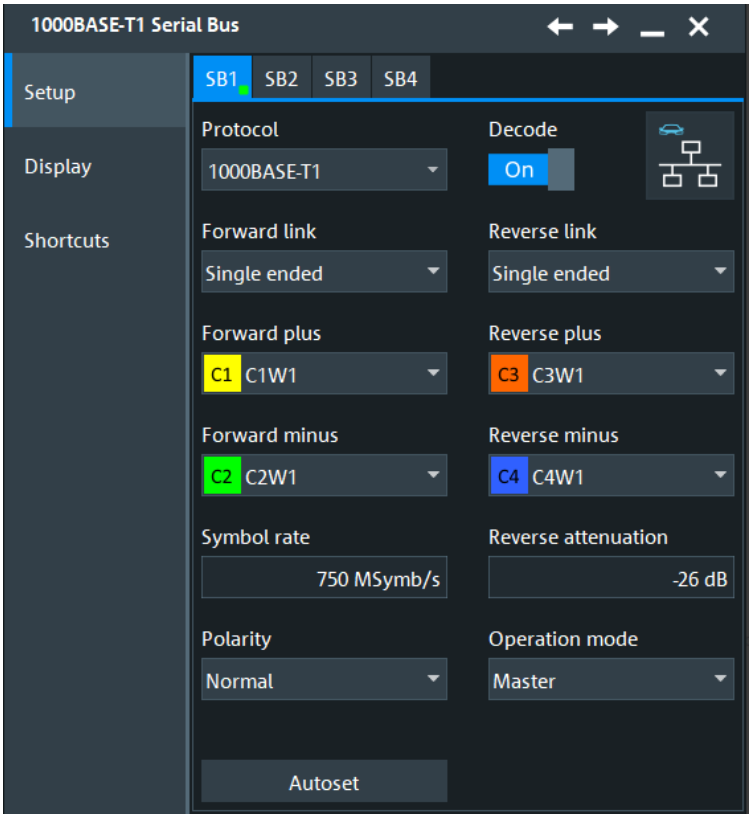
- Adopts full duplex communication of 1000BASE-T1 and therefore echo cancellation on a single twisted-pair channel. This feature reduces cabling while preserving the Ethernet MAC compatibility
- Adopts Pulse Amplitude Modulation (PAM-3) encoding scheme with the following encoding techniques:
  - PAM-3 is transmitted at 750 MB data rate.
  - It uses 15-bit scrambler to improve the EMC performance.
  - Data is encoded in 80B/81B blocks.
  - 396 bits of Reed-Solomon forward error correction (FEC) parity is added to each group of 45 80B/81B blocks.
  - 9-bit OAM symbol is added, then 396 bits Reed-Solomon. Together, they shape a PHY frame of 4050 bits, or 450 forward error correction (FEC) symbols of 9 bits. FEC allows for correction of up to 22 FEC symbols per PHY frame.

13.19.2 1000BASE-T1 configuration

13.19.2.1 1000BASE-T1 configuration settings

Access: [Bus] > "1000BASE-T1" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Forward
Defines the forward signal settings.

You can choose between a differential and single-ended signal and set the source channel for the waveform. Only analog channels are available for selection.

Remote command:
`BUS<m>:TBTO:FTYP` on page 2117
`BUS<m>:TBTO:FDIF` on page 2117
`BUS<m>:TBTO:FDAP` on page 2117
`BUS<m>:TBTO:FDAM` on page 2116

Reverse
Defines the reversed signal settings. The reverse direction is optional and can be disabled.

You can choose between a differential and single-ended signal and set the source channel for the waveform. Only analog channels are available for selection.

Remote command:
`BUS<m>:TBTO:RTYP` on page 2118
`BUS<m>:TBTO:RDIF` on page 2118
`BUS<m>:TBTO:RDAP` on page 2118
`BUS<m>:TBTO:RDAM` on page 2117

Reverse attenuation
Sets the attenuation factor. It is used to de-amplify the reverse signal before subtracting it from the forward signal.

The effective signal amplitude passed to the decoder is:

$$Amp = (Fwd_{+} - Fwd_{-}) - 10^{\frac{Attn}{20}}(Rev_{+} - Rev_{-})$$

Remote command:
`BUS<m>:TBTO:ATTN` on page 2116

Polarity
Selects the polarity of the data signal. You can select between "Normal" and "Inverted".

For "Normal" polarity, active high is defined as +1 V and active low is defined as -1 V.
For "Inverted" polarity, active high is defined as -1 V and active low is defined as +1 V.

Remote command:
`BUS<m>:TBTO:POLarity` on page 2118

Operation mode
Selects the desired direction of the full-duplex signal for analysis. If set to "Auto", the decoder automatically detects the "Master" or "Slave" mode.

Remote command:
`BUS<m>:TBTO:MODE` on page 2118

Autoset
Starts software algorithms for determining the signal threshold levels and bitrate.

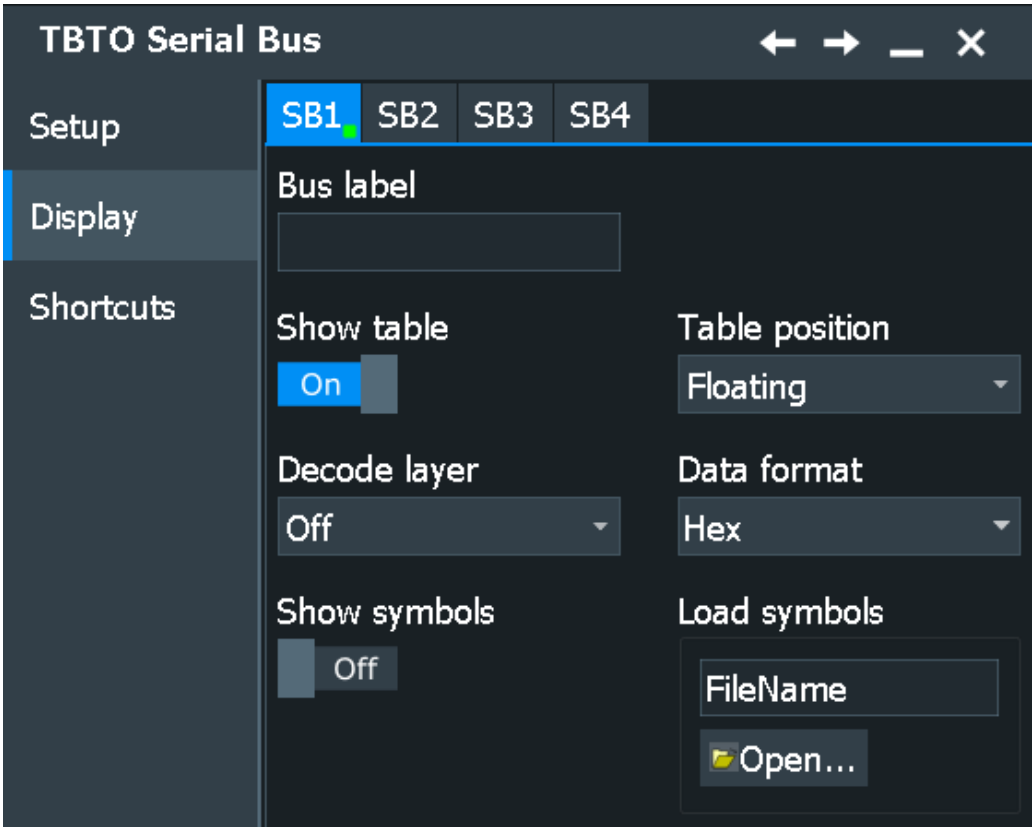
Remote command:
`BUS<m>:FAUToset` on page 1671

13.19.2.2 1000BASE-T1 display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

To enhance the decode possibilities of the 1000BASE-T1 protocol, you can use an additional setting in the "Display" tab: "Decode layer".



1000BASE-T1 Specific Label Settings

Label lists are protocol-specific. Label lists for 1000BASE-T1 are available in CSV and PTT format.

A 1000BASE-T1 label file contains two values for each ethernet header:

- Ethernet header display
- Symbolic label: name of the address, specifiing its function in the bus network.

Decode layer
Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

"Off"	No decode layer is displayed.
"Ternary symbols"	Decoded ternary symbols.
"Scrambled bits"	Scrambled ternary bits.
"Descrambled bits"	Descrambled ternary bits.
"Corrected RS-FEC symbols"	Displays RS-FEC symbols that have been corrected.

13.19.2.3 Configuring 1000BASE-T1 signals

For configuration, assign the lines to the input channels and define the active states and the logical thresholds.

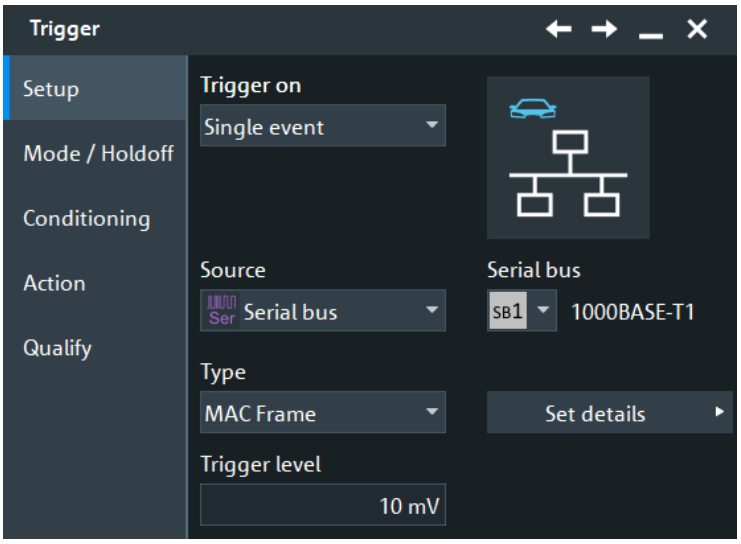
Serial bus setup

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. In the "Setup" tab, select the protocol: "1000BASE-T1".
4. Select the type and source for the reversed and forward signal.
5. Select the "Polarity" ("Normal" or "Inverted") of the signal.
6. Enable "Decode".

13.19.3 1000BASE-T1 trigger

13.19.3.1 1000BASE-T1 trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Type
Selects the trigger type for 1000BASE-T1 analysis.

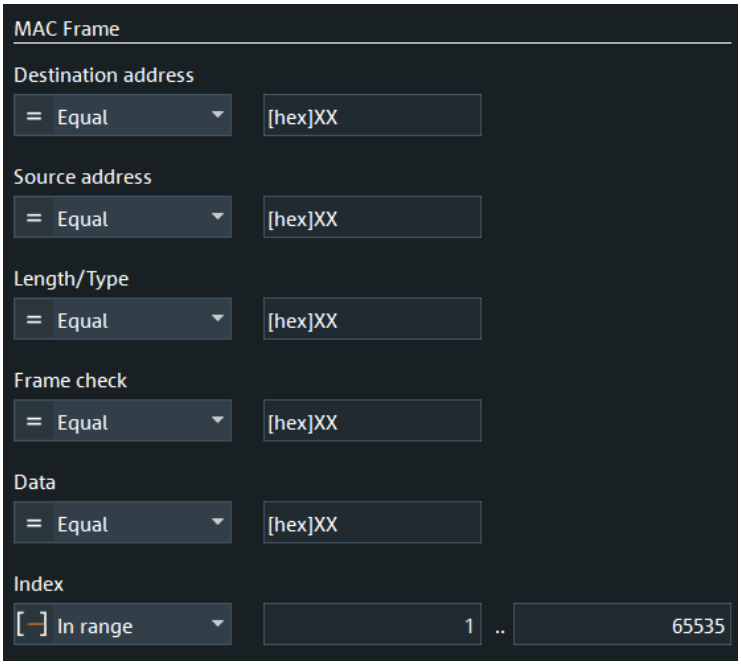
Remote command:
`TRIGger<m>:TBTO:TYPE` on page 2126

Frame Start ← Type
Triggers on the start of frame.

MAC Frame ← Type
Triggers on a media access control (MAC) frame. You can specify:

- Destination address
- Source address
- Length/Type
- Frame check
- Data
- Index

The frame contains addresses of the devices and MAC control information that define how to go about transmitting and receiving frames.



IDLE ← Type
Triggers on an idle frame. The frame is used for clock synchronization.

Error ← Type
Triggers on the specified error frame.

Trigger level
Sets the trigger level.

Remote command:
`TRIGger<m>:TBTO:TRGLevel` on page 2126

Destination address
Sets the specified destination address to be triggered on. The destination address setup consists of the condition and one or two data patterns.

The destination address corresponds to the address of the interface in the device that receives the frame. If the destination address does not match the interface's own Ethernet address, then the interface is free to ignore the rest of the frame.

"Condition" Defines the operator to set a specific destination address, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the destination address pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:TBTO:DADDress:CONDition` on page 2127
`TRIGger<m>:TBTO:DADDress:MIN` on page 2128
`TRIGger<m>:TBTO:DADDress:MAX` on page 2128

Source address
Sets the specified source address to be triggered on. The source address setup consists of the condition and one or two data patterns.

The source address is the physical address of the device that sends the frame.

"Condition" Defines the operator to set a specific source address, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the source address pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:TBTO:SADDress:CONDition` on page 2132
`TRIGger<m>:TBTO:SADDress:MIN` on page 2133
`TRIGger<m>:TBTO:SADDress:MAX` on page 2133

Length/Type
Sets the specified length/type to be triggered on. The length/type setup consists of the condition and one or two data patterns.

The value in this field indicates the manner in which the field is being used, either as a length or type field. As a length field, the value in the field indicates the number of logical link control (LLC) data octets that follow in the data field of the frame. As a type field, the value in the field is used to indicate the type of protocol data being carried in the data field of the frame.

"Condition" Defines the operator to set a specific Length/Type, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the Length/Type pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:TBTO:LENGth:CONDition` on page 2131
`TRIGger<m>:TBTO:LENGth:MIN` on page 2132
`TRIGger<m>:TBTO:LENGth:MAX` on page 2132

Frame check

Sets the specified frame check to be triggered on. The frame check setup consists of the condition and one or two data patterns.

This field contains a value that is used to check the integrity of the various bits in the frame fields (excluding the preamble / SFD).

"Condition" Defines the operator to set a specific frame check, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the frame check pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:TBTO:CRC:CONDition` on page 2126
`TRIGger<m>:TBTO:CRC:MIN` on page 2127
`TRIGger<m>:TBTO:CRC:MAX` on page 2127

Data

Sets the specified data to be triggered on. The data setup consists of the condition and one or two data patterns.

"Condition" Defines the operator to set a specific data, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the data pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:TBTO:DATA:DCONdition` on page 2128
`TRIGger<m>:TBTO:DATA:DMIN` on page 2129
`TRIGger<m>:TBTO:DATA:DMAX` on page 2129

Index

Sets the specified value or range within this series of data that is considered for the analysis. The index setup consists of the condition and one or two index values.

"Condition" Defines the operator to set a specific index, e.g. "Equal" or a range.

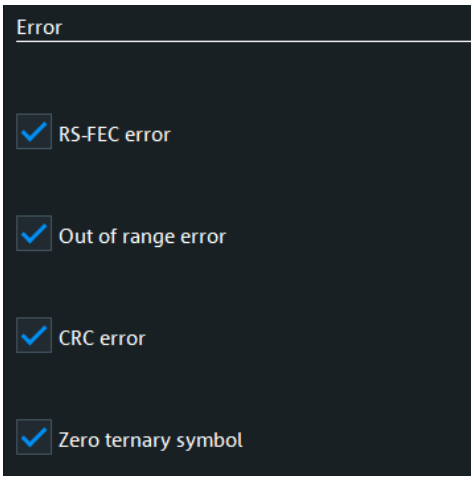
"Index Min/Index"
Defines the bit pattern of the index pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Index Max" The second index pattern is required to specify a range with conditions "In range".

Remote command:
`TRIGger<m>:TBTO:DATA:ICONdition` on page 2129
`TRIGger<m>:TBTO:DATA:IMIN` on page 2130
`TRIGger<m>:TBTO:DATA:IMAX` on page 2130

Errors
Sets the type of error events to be triggered on.



RS-FEC error ← Errors
Triggers, if more than 22 RS-FEC errors are detected.

Remote command:
`TRIGger<m>:TBTO:ERRor:FEC` on page 2131

CRC error ← Errors
Triggers on a frame that has a mismatch of the cyclic redundancy check (CRC) value between the transmitting and receiving device.

Remote command:
`TRIGger<m>:TBTO:ERRor:CRC` on page 2130

Out of range error ← Errors
Triggers if values are out of range. This can be, for example a preamble error, or invalid symbols in control blocks.

Remote command:
`TRIGger<m>:TBTO:ERRor:OOR` on page 2131

Zero terrnary symbol ← Errors
Triggers, if a zero ternary symbol pair (0-0) is detected.

Remote command:
`TRIGger<m>:TBTO:ERRor:ZERO` on page 2131

13.19.3.2 Triggering on 1000BASE-T1

Prerequisite: A bus is configured for the 1000BASE-T1 signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to 1000BASE-T1.
5. Tap "Type" and select the trigger type to be used for 1000BASE-T1 protocol analysis.
6. To refine the trigger settings, configure additional settings, which are available for some trigger types.

For details, see Chapter 13.19.3, "1000BASE-T1 trigger", on page 798.

13.19.4 1000BASE-T1 decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Serial Bus" dialog > "Setup" tab, enable "Decode".
2. In the "Serial Bus" dialog > "Display" tab, select additional result display settings: "Show table" and "Show binary".
3. If required, enable "Zoom coupling"

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The example shows decoded signals in 1000BASE-T1.

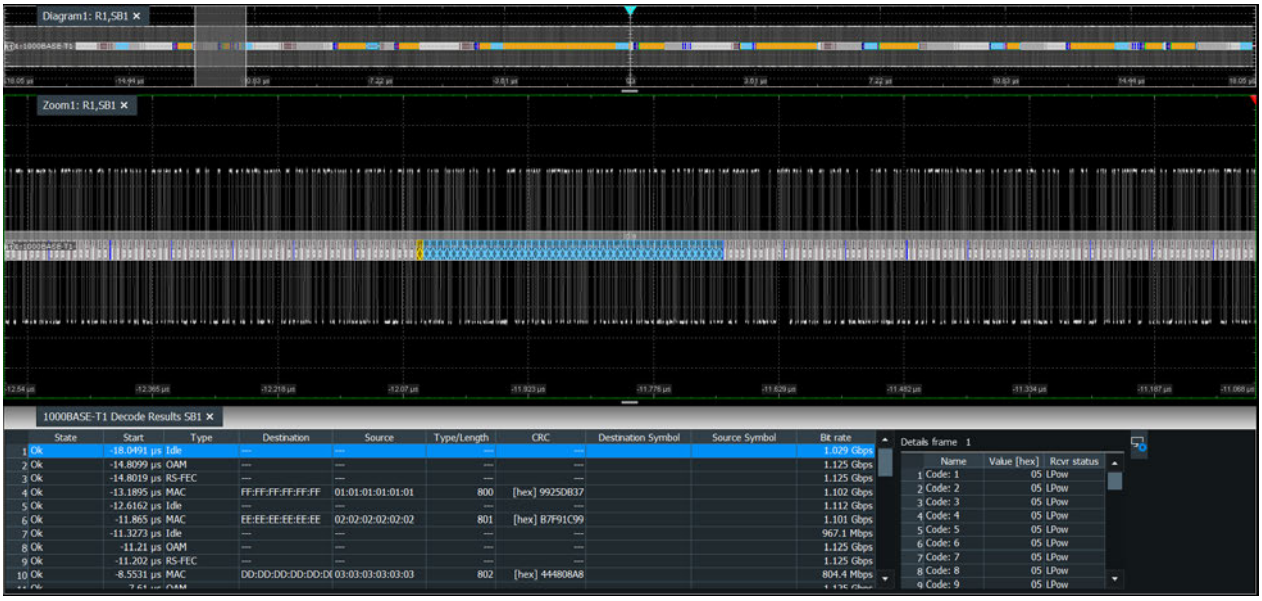


Figure 13-34: 1000BASE-T1 decoded signal

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

Table 13-30: Content of the Decode results table

Column	Description
Frame	Frame count
State	Overall state of the frame: either OK or the relevant error condition (preamble, CRC or SFD)
Start	Start time of the frame
Type	Type of frame (e.g. Idle, MAC or data)
Destination	Destination symbol of the frame
Source	Source address of the frame
Type/Length	The sub-protocol (e.g. HTML, video, etc.) determines what meaning this field has. Since the content of this data area is not decoded, the interpretation of this field is ambivalent. It could either be the word type (specific for the sub-protocol) or the word length.
CRC	Frame Check (Cyclic Redundancy Code, CRC)
Destination Symbol	Destination address of the frame
Source Symbol	Source symbol of the frame
Data	Values of the data bytes in a frame. The table shows a truncated version; to see all the bytes in a separate data table, activate "Show details". The data format is always hexadecimal.

Column	Description
CRC	Frame Check (Cyclic Redundancy Code, CRC)
Bit rate	Value of the bit rate

Table 13-31: Content of the Details frame table

Column	Description
Name	State of the field
Value [hex]	Hexadecimal value of the field
Rcvr status	Receiver status

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands are described in Chapter 24.17.20, "Ethernet 1000BASE-T1 (option R&S RTO6-K560)", on page 2116.

13.19.5 Search on decoded 1000BASE-T1 data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

Before you can start the search, you have to configure the bus correctly and acquire decoded data.

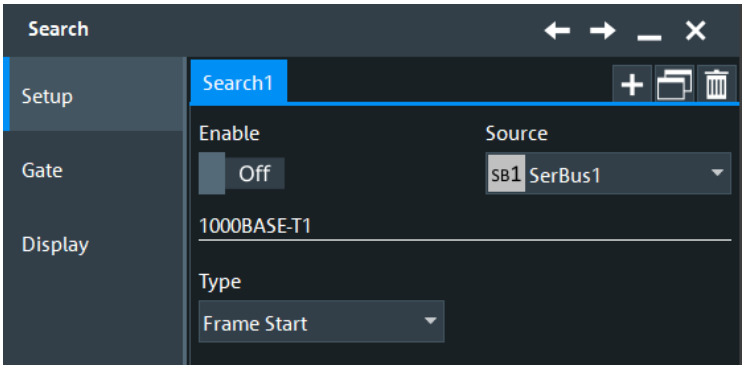
To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 11, "Search functions", on page 419.

13.19.5.1 1000BASE-T1 search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"

Search criteria
Use the "Search criteria" dialog to define the event types to be searched.



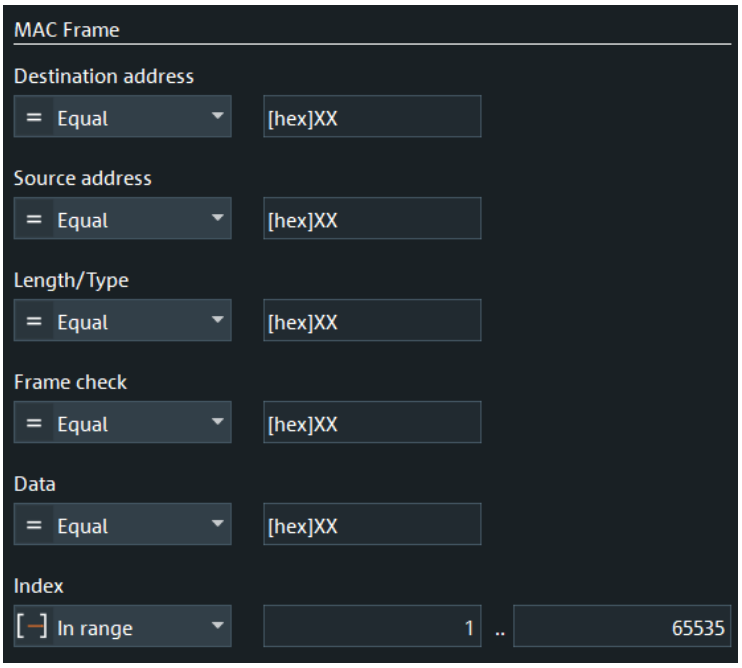
Type
Selects the search type for 1000BASE-T1 analysis.

Remote command:
`SEARch:TRIGger:TBTO:TYPE` on page 2141

Frame Start ← Type
Searches for the start of frame.

MAC Frame ← Type
Searches for a Media Access Control (MAC) frame that matches the specified Destination address, Source address, Length/Type, Frame check, Data or Index condition.

The frame contains addresses of the devices and MAC control information that define how to go about transmitting and receiving frames.



IDLE ← Type
Searches for an idle frame. The frame is used for clock synchronization.

Error ← Type
Searches for the specified error frame.

Destination address
Sets the specified destination address to be searched for. The destination address setup consists of the condition and one or two data patterns.

The destination address corresponds to the address of the interface in the device that receives the frame. If the destination address does not match the interface's own Ethernet address, then the interface is free to ignore the rest of the frame.

"Condition" Defines the operator to set a specific destination address, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the destination address pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:TBTO:DADDress:CONDition` on page 2135
`SEARch:TRIGger:TBTO:DADDress:MIN` on page 2136
`SEARch:TRIGger:TBTO:DADDress:MAX` on page 2135

Source address
Sets the specified source address to be searched for. The source address setup consists of the condition and one or two data patterns.

The source address is the physical address of the device that sends the frame.

"Condition" Defines the operator to set a specific source address, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the source address pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:TBTO:SADDress:CONDition` on page 2140
`SEARch:TRIGger:TBTO:SADDress:MIN` on page 2141
`SEARch:TRIGger:TBTO:SADDress:MAX` on page 2140

Length/Type
Sets the specified Length/Type to be searched for. The Length/Type setup consists of the condition and one or two data patterns.

The value in this field indicates the manner in which the field is being used, either as a length or type field. As a length field, the value in the field indicates the number of logical link control (LLC) data octets that follow in the data field of the frame. As a type field, the value in the field is used to indicate the type of protocol data being carried in the data field of the frame.

"Condition" Defines the operator to set a specific Length/Type, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the Length/Type pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:TBTO:LENGth:CONDition` on page 2139
`SEARch:TRIGger:TBTO:LENGth:MIN` on page 2140
`SEARch:TRIGger:TBTO:LENGth:MAX` on page 2139

Frame check
Sets the specified frame check to be searched for. The frame check setup consists of the condition and one or two data patterns.

This field contains a value that is used to check the integrity of the various bits in the frame fields (excluding the preamble / SFD).

"Condition" Defines the operator to set a specific frame check, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the frame check pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:TBTO:CRC:CONDition` on page 2134
`SEARch:TRIGger:TBTO:CRC:MIN` on page 2135
`SEARch:TRIGger:TBTO:CRC:MAX` on page 2134

Data
Sets the specified data to be searched for. The data setup consists of the condition and one or two data patterns.

"Condition" Defines the operator to set a specific data, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the data pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:TBTO:DATA:DCONdition` on page 2136
`SEARch:TRIGger:TBTO:DATA:DMIN` on page 2137
`SEARch:TRIGger:TBTO:DATA:DMAX` on page 2136

Index
Sets the specified value or range within this series of data that is considered for the search. The index setup consists of the condition and one or two index values.

"Condition" Defines the operator to set a specific index, e.g. "Equal" or a range.

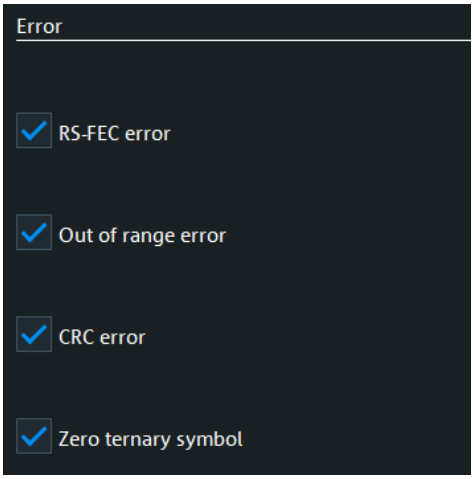
"Index Min/Index"
Defines the bit pattern of the index pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Index Max" The second index pattern is required to specify a range with conditions "In range".

Remote command:
`SEARch:TRIGger:TBTO:DATA:ICONdition` on page 2137
`SEARch:TRIGger:TBTO:DATA:IMIN` on page 2138
`SEARch:TRIGger:TBTO:DATA:IMAX` on page 2137

Errors
Sets the type of error events to be searched for.



RS-FEC error ← Errors
Searches for RS-FEC errors.

Remote command:
`SEARch:TRIGger:TBTO:ERRor:FEC` on page 2138

CRC error ← Errors
Searches for a frame that has a mismatch of the Cyclic Redundancy Check (CRC) value between the transmitting and receiving device.

Remote command:
`SEARch:TRIGger:TBTO:ERRor:CRC` on page 2138

Out of range error ← Errors
Searches for values that are out of range. This can be, for example a preamble error, or invalid symbols in control blocks.

Remote command:
`SEARch:TRIGger:TBTO:ERRor:OOR` on page 2138

Zero terrnary symbol ← Errors
Searches for zero ternary symbol pairs (0-0).

Remote command:
`SEARch:TRIGger:TBTO:ERRor:ZERO` on page 2139

13.19.5.2 1000BASE-T1 search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands:

- `SEARch:RESult:TBTO:FCOunt?` on page 2142
- `SEARch:RESult:TBTO:FRAMe<m>:TYPE?` on page 2146
- `SEARch:RESult:TBTO:FRAMe<m>:STATe?` on page 2145
- `SEARch:RESult:TBTO:FRAMe<m>:STARt?` on page 2145
- `SEARch:RESult:TBTO:FRAMe<m>:STOP?` on page 2145
- `SEARch:RESult:TBTO:FRAMe<m>:DESTaddress?` on page 2142
- `SEARch:RESult:TBTO:FRAMe<m>:SRCaddress?` on page 2144
- `SEARch:RESult:TBTO:FRAMe<m>:DATA?` on page 2142
- `SEARch:RESult:TBTO:FRAMe<m>:CRC?` on page 2142
- `SEARch:RESult:TBTO:FRAMe<m>:NUMWords?` on page 2144
- `SEARch:RESult:TBTO:FRAMe<m>:FTYPe?` on page 2143
- `SEARch:RESult:TBTO:FRAMe<m>:DSYMbol?` on page 2143

- `SEARch:RESult:TBTO:FRAMe<m>:SSYMbol?` on page 2144
- `SEARch:RESult:TBTO:FRAMe<m>:WORD<n>:TYPE?` on page 2146
- `SEARch:RESult:TBTO:FRAMe<m>:WORD<n>:VALue?` on page 2146
- `SEARch:RESult:TBTO:FRAMe<m>:WORD<n>:VSTR?` on page 2147

13.19.5.3 Searching 1000BASE-T1 data

Prerequisite: A serial bus is configured for the 1000BASE-T1 signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Tap "Setup search".
3. If the dialog box does not contain a search entry, tap the "Add" icon to create one, as described in "To create a user-defined search" on page 434.
4. Tap "Source" and select the serial bus that is set to 1000BASE-T1 (e.g. "SB1", unless already selected).

   The search dialog for 1000BASE-T1 protocol analysis opens.
5. Specify search criteria according to Chapter 13.19.5, "Search on decoded 1000BASE-T1 data", on page 806.
6. To acquire a waveform, press [Single].

   The R&S RTO6 performs an 1000BASE-T1 decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).
7. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog.

   The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.20 USB (option R&S RTO6-K570)

R&S RTO6-K570 is a firmware option that enables the R&S RTO6 to analyze Universal Serial Bus (USB) protocols, by triggering and decoding them. The option is compatible with the standards USB 1.0, USB 1.1, USB 2.0 and USB HSIC (High-Speed Inter-Chip). R&S RTO6-K570 supports the data rates "Low Speed" (1.5 Mbit/s), "Full Speed" (12 Mbit/s) and "High Speed" (480 Mbit/s, available in USB 2.0 and HSIC).

- The USB protocol....813
- USB 2.0 configuration....818
- USB 2.0 trigger....824
- USB 2.0 decode results....832
- Search on decoded USB 2.0 data....833

13.20.1 The USB protocol

The USB protocol was developed, starting in 1996, by the nonprofit organization USB Implementers Forum, Inc. (USB-IF), formed by Compaq, Hewlett-Packard, Intel, Lucent Technologies, Microsoft, NEC, and Philips. The purpose was to provide a common "plug-and-play" solution to replace a multitude of interfaces for the communication between computers and devices. It should allow even unskilled users to easily connect many devices to a PC. USB was originally used for devices that feature low signalling rates (up to 1.5 Mbit/s), especially human interface devices like mouse, joystick or keyboard. With the release of USB 2.0 in April 2000, devices such as printers, cameras and mass storage media were enabled to exchange data at faster rates (up to 480 Mbit/s). Also, the high-speed protocol HSIC (High Speed Inter Chip) is used for the communication between on-board devices.

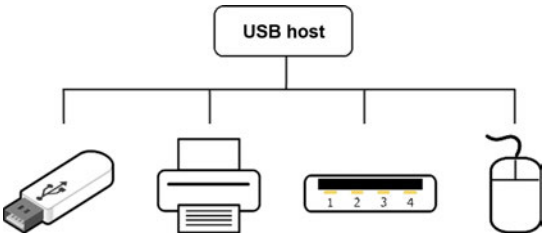


Figure 13-35: USB Topology: mass storage device, printer, USB hub, and mouse (human interface device) connected to a USB host

More information, including the USB specification, is available online within the web domain http://usb.org.

This chapter provides an overview of protocol characteristics, data transfer, packet structure, address and endpoint formats of USB and trigger possibilities.

USB characteristics

Main characteristics of USB are:

- Four-wire design: USB requires a shielded cable containing four wires. Two of them, called D+ and D-, form a twisted pair (for low speed, they may not be twisted). These data lines transmit differential data signals and single-ended signal states, both referenced to a third wire: the GND or ground. The fourth wire, called VBUS (voltage bus), carries a nominal 5 V supply, which may be used to power a device.
- Host-to-device communication: in USBs "speak-when-spoken-to" protocol, communication is always initiated by the host. Consequently, there is no direct communication between USB devices, apart from few exceptions.

- Addressing scheme: a maximum of 127 connected devices can be distinguished, because a packet's address field length is limited to 7 bits. USB devices have up to 16 OUT endpoints (from host to device) and up to 16 IN endpoints (from device to host).
- USB transactions consist of two or three packets: token, data, and typically handshake
- Packet type: a packet identifier (PID) is sent as a first byte within the packet and specifies the different packet types.
- NRZI (Non Return to Zero Inverted): a zero (0) is encoded as a transition of the physical level, whereas a one (1) has no transition, thus it is represented by a steady level.

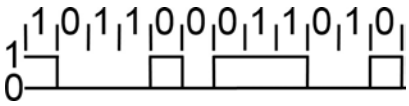


Figure 13-36: Example of an NRZI sequence

- Bit stuffing: a zero (0) is inserted after every 6 consecutive ones (111111). This ensures sufficient transitions to keep the phase-locked loop (PLL) synchronized and locked onto the data stream.
- Little Endian scheme: both multiple bits and multiple bytes are transmitted with the least significant bit/byte (LSB) sent first, while the most significant bit/byte (MSB) is sent last.
- HSIC (High-Speed Inter-Chip): an industry standard for USB chip-to-chip interconnection with a 2-signal (strobe, data) source synchronous serial interface, using 240 MHz DDR signaling to provide only high-speed (480 Mbps) data rate.

Table 13-32: Simplified symbolic representation from the USB standard

Bus State	Protocol	Levels
Differential **"1"**		D+ High, D- Low
Differential **"0"**		D+ Low, D- High
Data **"J"** State	Low speed	D+ Low, D- High (differential **"0"**)
	Full speed	D+ High, D- Low (differential **"1"**)
Data **"K"** State	Low speed	D+ High, D- Low (differential **"1"**)
	Full speed	D+ Low, D- High (differential **"0"**)

In the example in Table 13-32, High speed can be assumed to be like Full speed. For a complete overview of bus states and for an electrical definition of High and Low speed, refer to chapter 7 of the USB specification.

Data transfer

In contrast to a conventional bus, USB is more like a network protocol, using target addresses and endpoints. However, USB features a bus master, called the host. It transmits packets of data to all devices or hubs connected to the host (or connected to a device or hub, which is in turn connected to the host). All connected devices receive

each data packet, but disregard it unless it carries the correct address. In reply, only the addressed device (one at a time) can send data upstream, to the host.

The USB architecture supports four different kinds of data transfer: control, bulk, interrupt or isochronous transfer. For example, a "bulk OUT" transfer (from host to device) would look like this:

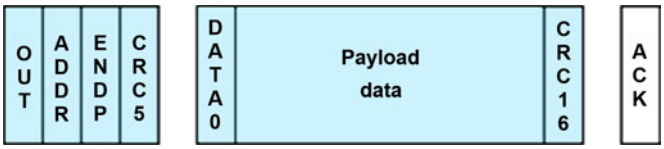


Figure 13-37: Example of a bulk OUT transfer. Blue: host speaks, white: device speaks

Packet structure

All packets must start with a **SYNC** field, also called **SOP** (start of packet), which indicates data transmission. It consists of "KJ" pairs, followed by one "KK". At low speed and full speed (USB 1.x), it is 8 bits long, encoded as "KJKJKJKK". At high speed (USB 2.0 and HSIC), it is up to 32 bits long, encoded as "KJKJKJKJKJKJ...KK".

The SYNC field is used to synchronize the clock of the receiver with that of the transmitter. The final 2 bits ("KK") indicate where the PID fields starts.

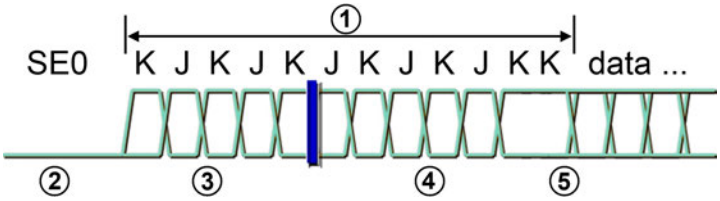


Figure 13-38: Start of a high-speed packet: after an SE0, the packet starts with a SYNC field (or SOP), followed by the data packet

SE0 = Single-ended zero, both D+ and D- wire are on low level
1 = High-speed SYNC field (or SOP), length up to 32 bits
2 = Receiver-squelched differential envelope, below 100 mV
3 = Differential envelope exceeds 150 mV, reception enabled with 4 symbol times
4 = Clock recovery must lock in time to detect end of SYNC
5 = End of SYNC detected, receiver begins data recovery

After the SYNC (or SOP), USB data packets consist of (in this order):

- **PID** (packet identifier): specifies the type of packet. Examples are:
  - Token packets (IN, OUT, SETUP) are always the first packet in a USB transaction, containing the address of the device and its endpoint.
  - Token SOF packets (start of frame) are there to schedule a data transfer.
  - Handshake packets provide the status of the transaction.
  - Data packets are dedicated for the data transfer payload.
  - Various special packets provide special characteristics: for instance, SPLIT packets are used to communicate with low or full speed devices on a high-speed link.

The PID's length is 8 bit, beginning (LSB) with the fields PID0, PID1, PID2 and PID3. Those 4 bits are repeated in PID4, PID5, PID6 and PID7, but in complementary form (for integrity checking). The PID is the only part of a packet not to be covered by the CRC (see below). Packets with invalid or non-supported PID fields are rejected. For valid PIDs, see Table 13-33.

- **ADDR** (address): contains the USB address of the target device. The length of this field is 7 bits, resulting in addresses 0 through 127 (with 0 reserved as the default address after reset).
- **ENDP** (endpoint): number of the endpoint. The length of this field is 4 bits, resulting in a maximum of 16 different endpoints.
- **Frame number**: this field with a length of 11 bits specifies the current frame number 0 to 2047. It is only sent in SOF packets.
- **Data field** (or "payload data"): the length of this field is not fixed, it can contain between 0 bytes and 1024 bytes.
- **CRC** (cyclic redundancy check): Finally, the CRC field ensures that the received data is correct. If an error is detected in the CRC calculation, the packet is rejected. Packets (in this case: IN, OUT, SOF, SETUP, PING, SPLIT) are followed by a CRC5 (which is 5 bits long) over ADDR and ENDP (or the frame number field, in case of SOF); data packets (DATA0, DATA1, MDATA, DATA2) are followed by a CRC16 (which is 16 bits long) over the data field. No CRC is used for handshake packets.
- **EOP** (end of package): the devices determine the length of the data field with the EOP signal, which indicates the end of a packet. In low and full speed, it consists of two SE0 bits, followed by a J state. For high speed, the current state is inverted and kept for 7-bit times, leading to a Bitstuff error that is used as an EOP indicator. This corresponds to the bit sequence 01111111 ahead of NRZI; for the SOF token, the bit sequence 0111111111111111111111111111111111111111 is transmitted (five NRZI bytes), which is used to recognize disconnects (since high-speed devices normally do not apply voltage to D+/D-). If no device is present, a higher voltage is measured. After that, the idle state (SE0) is occupied.

Table 13-33: Valid PIDs for different packet types

Packet type	Name	PID value	Meaning
Token	OUT	0xE1	Starts data transfer towards a device
	IN	0x69	Starts data transfer towards the host
	SOF	0xA5	Indicates start of frame
	SETUP	0x2D	Starts a setup transfer and sends information on this to the device
Data	DATA0	0xC3	Data packet with data-toggle bit 0
	DATA1	0x4B	Data packet with data-toggle bit 1
	DATA2	0x87	Data packet for high speed IN isochronous transfers that require high bandwidth
	MDATA	0x0F	Data packet for high-speed OUT isochronous transfers that require high bandwidth
Handshake	ACK	0xD2	Acknowledgement of a packet received without error

Packet type	Name	PID value	Meaning
	NAK	0x5A	Data not accepted, typically equivalent with some type of EAGAIN, meaning that the data should be resent later
	STALL	0x1E	A severe error has occurred, the target endpoint cannot be addressed until it is explicitly cleared again
	NYET	0x96	Only used in high-speed transfers, meaning ACK, but in the next interval no data can be received, therefore the host should first apply a PING
Special	PRE	0x3C	Starts a low speed transfer via a full speed bus
	ERR	0x3C	Indicates an error in an SPLIT transaction (using the same PID as PRE, however, these cannot be mistaken for each other)
	SPLIT	0x78	Starts an SPLIT transaction (thus: a low speed or full speed transfer via a high-speed bus)
	PING	0xB4	Used for monitoring high-speed data flow

- Token packets (IN, OUT, SETUP) and PING packets have the following format:

SYNC	PID	ADDR	ENDP	CRC5	EOP

- Start of frame packets (SOF) have the following format:

SYNC	PID	Frame number	CRC5	EOP

- Data packets have the following format:

SYNC	PID	Data	CRC16	EOP

- Handshake packets have the following format:

SYNC	PID	EOP

- SPLIT packets have the following format:

SYNC	PID	ADDR	SC	PORT	S	E	ET	CRC5	EOP

Trigger possibilities

Signals on the input channels CH1 - CH4 of the R&S RTO6 can be triggered by the option R&S RTO6-K570. The following trigger types are available:

- ANY Packet - Packet Sync: Triggering on the first rising slope after transmission of the packet Sync. Various lengths according to standard (in USB 2.0: 32 bit)
- Any token, OUT, IN, SOF, SETUP, AND-ing with user defined PID check, address, endpoint, CRC5: For OUT, IN, SETUP, the endpoint and CRC5 follow from the bit order, therefore such patterns can be recognized.

- Data Selection: DATA0, DATA1, DATA2, MDATA, (for USB 1.x only: AND-ing with user defined PID check, payload and CRC values)
- Handshake Packet Setup: Triggering on handshake packet, trigger with specific settings: ACK, NAK, NYET, STALL or ERR handshake packet
- Protocol Error: Triggering on PID/check error, CRC5 error, CRC16 (for USB 1.x only), frame length error (for USB 1.x only)
- Bus Event: Triggering on reset, resume, or suspend

13.20.2 USB 2.0 configuration

If you need information on how to get started with configuring the USB 2.0 setup, see Chapter 13.20.2.2, "Configuring USB signals", on page 823. Otherwise proceed with the configuration settings.

13.20.2.1 USB 2.0 configuration settings

Access: [Bus] > "USB 2.0" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.

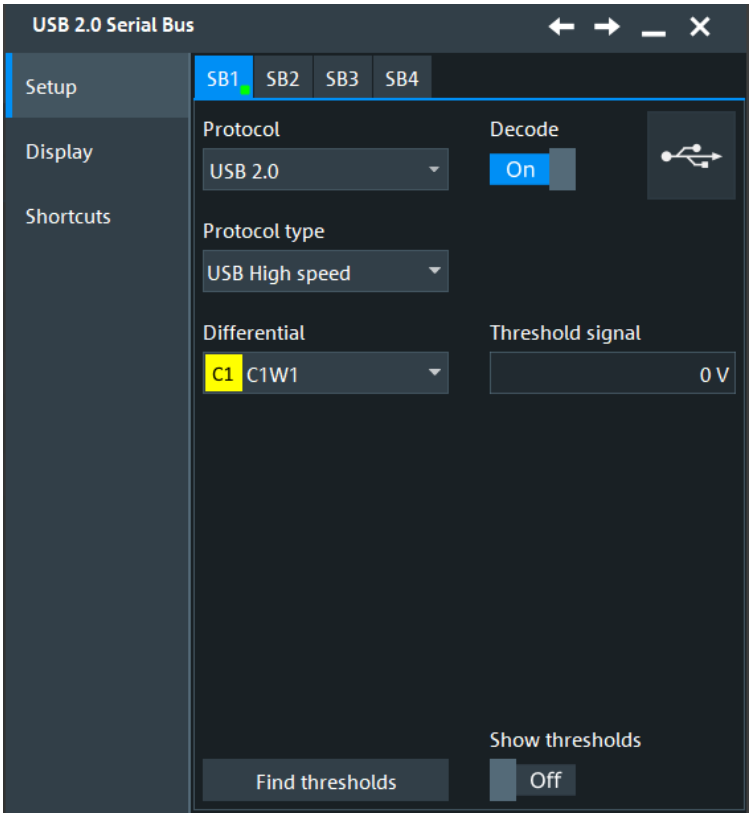


Figure 13-39: Serial bus protocol configuration dialog

See also: Chapter 13.1.1, "Setup - general settings", on page 485

Protocol type
Defines the USB protocol technology and transmission speed.

"USB Low speed"
Selects USB 2.0 low speed protocol (1.5 Mbit/s).

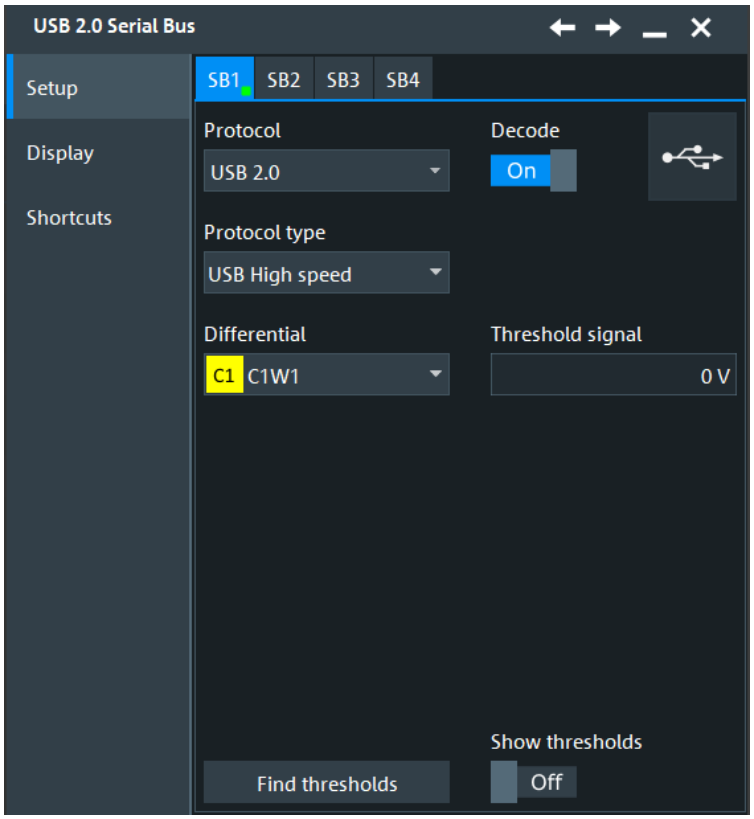


Figure 13-40: USB low speed protocol configuration

"USB Full speed"

Selects USB 2.0 full speed protocol (12 Mbit/s).

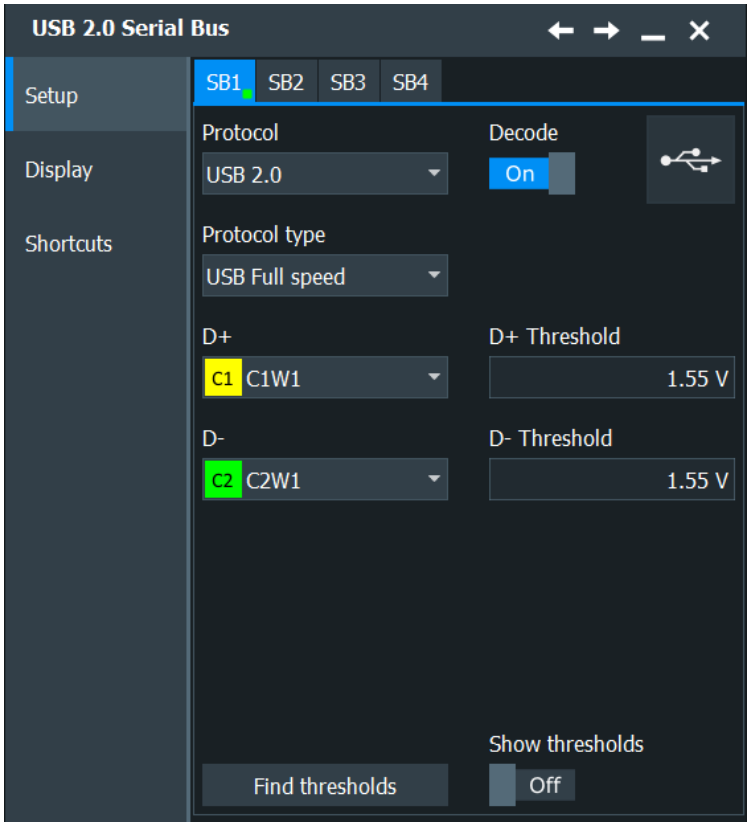


Figure 13-41: USB full speed protocol configuration

"USB High speed"

Selects USB high-speed protocol (480 Mbit/s). As the signal is differential, there is only one source and one threshold to be defined.

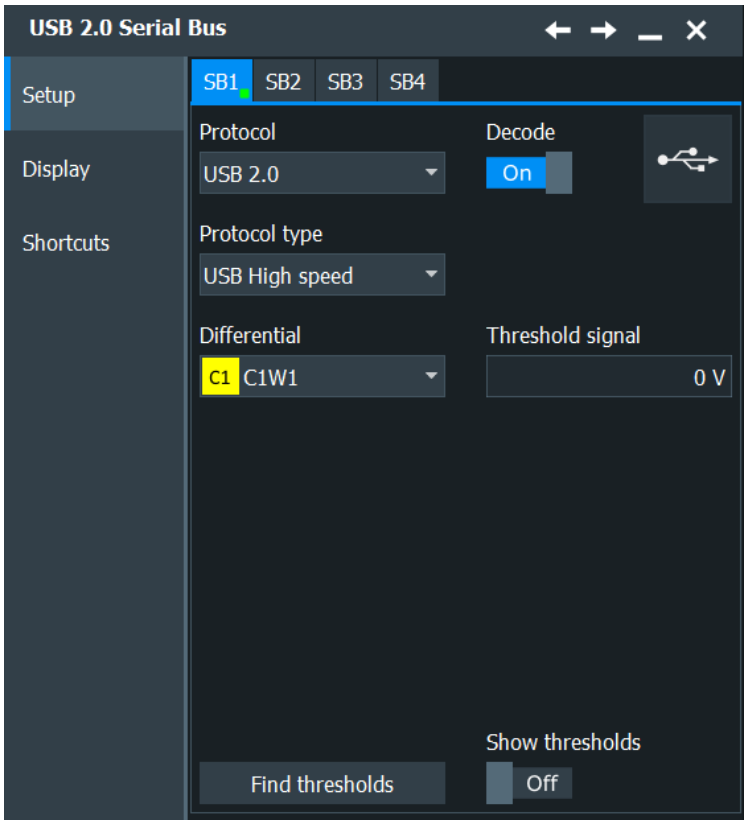


Figure 13-42: USB high-speed protocol configuration

"USB HSIC"

Selects USB high speed inter-chip (HSIC) protocol.

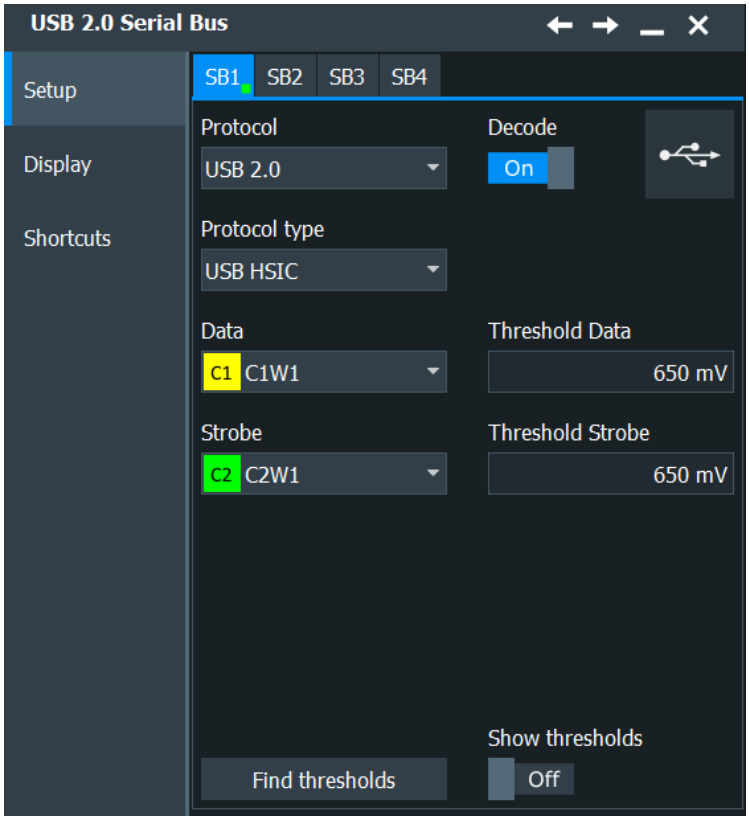


Figure 13-43: USB HSIC protocol configuration

Remote command:
`BUS<m>:USB:TECHnology` on page 2148

D+, D-

Define the source settings for the D+ and the D- data signals for USB low speed and USB full speed protocols.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Remote command:
`BUS<m>:USB:DPLus:SOURce` on page 2148
`BUS<m>:USB:DMINus:SOURce` on page 2148

Differential

Defines the source settings for the differential data signal (in USB high-speed protocol, only). You can select analog channels "C1" to "C4", and also active math and reference waveforms as source. If you want to trigger on the serial bus, analog channel sources are required.

Remote command:
`BUS<m>:USB:DIFFerential:SOURce` on page 2149

Data, Strobe

Define the source settings for the data and the strobe signals in USB HSIC protocol. You can select analog channels "C1" to "C4", and also active math and reference waveforms as source. If you want to trigger on the serial bus, analog channel sources are required.

Remote command:

`BUS<m>:USB:DATA:SOURce` on page 2149

`BUS<m>:USB:STRobe:SOURce` on page 2149

Thresholds

Sets the threshold value for the digitization of each signal line. If the signal value on the line is higher than the threshold, the signal state is high. Otherwise, if the signal value is below the threshold, the signal state is considered low.

"Threshold"	Enter the value directly in the field.
"Find thresholds"	Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:

`BUS<m>:USB:DPLus:THReshold` on page 2149

`BUS<m>:USB:DMINus:THReshold` on page 2150

`BUS<m>:USB:DIFFerential:THReshold` on page 2150

`BUS<m>:USB:DATA:THReshold` on page 2150

`BUS<m>:USB:STRobe:THReshold` on page 2150

`BUS<m>:SETReflevels` on page 1671

13.20.2.2 Configuring USB signals

For configuration, assign the lines to the input channels and define the active states and the logical thresholds.

For details on configuration settings, see Chapter 13.20.2.1, "USB 2.0 configuration settings", on page 818.

Serial Bus Setup

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. In the "Setup" tab, select "Protocol" = "USB 2.0".
4. Tap "Protocol type" and select the protocol type that you want to set up.
5. Depending on the protocol type, select the waveform for each of the available "D+", "D-", "Differential", "Data", and "Strobe" lines.
6. Set the logical thresholds: Either to the middle reference levels with "Find thresholds", or enter a user-defined value directly in the "Threshold" fields.
7. Enable "Decode".

13.20.3 USB 2.0 trigger

If you need information on how to get started with triggering on USB 2.0 signals, see Chapter 13.20.3.2, "Triggering on USB 2.0", on page 831. Otherwise proceed with the USB 2.0 trigger settings.

13.20.3.1 USB 2.0 trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"

In this section, all trigger settings are described. Their availability on the instrument depends on the selected USB 2.0 protocol type and trigger type. The user interface of the instrument displays only appropriate settings and guides you through the trigger setup.

For a list of supported trigger conditions, refer to data sheet.

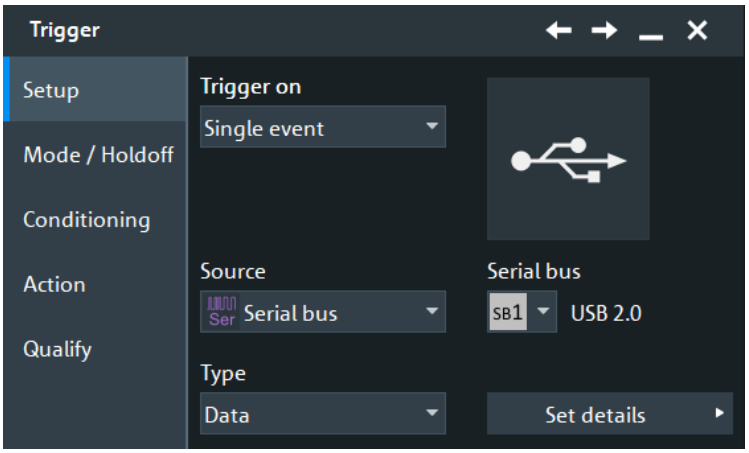


Figure 13-44: USB trigger event settings dialog

Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Trigger type USB

Selects the trigger type for USB 2.0 analysis. The available trigger types depend on the USB 2.0 protocol type that is selected in the configuration setup, see "Protocol type" on page 819.

Remote command:
`TRIGger<m>:USB:TYPE` on page 2152

Start of packet ← Trigger type USB
Sets the trigger to the SOP (start of packet) field. The start of packet condition is the end of the SYNC field. The trigger instant is the end of the SOP field.

End of packet ← Trigger type USB
Sets the trigger to the EOP (end of packet) field. The trigger instant is the beginning of the EOP field.

Reset ← Trigger type USB
Sets the trigger to the Reset field. For more information on the reset condition, see the USB standard. The trigger instant is the end of the 10 ms period after the SE0 field.

Suspend ← Trigger type USB
Sets the trigger to the Suspend field. For more information on the suspend condition, see the USB standard. The trigger instant will be declared after the defined 3 ms time-out.

Resume ← Trigger type USB
Sets the trigger to the Resume field. For more information on the resume condition, see the USB standard. The trigger instant will be declared after the defined 20 ms timeout.

Token ← Trigger type USB
Sets the trigger to one out of four different token trigger types: OUT, IN, SOF, or SETUP.

See "Token" on page 826

Data ← Trigger type USB
Sets the trigger to one out of four different data trigger types: DATA0, DATA1, DATA2, or MDATA.

See "Data" on page 826

Handshake ← Trigger type USB
Sets the trigger to one out of four different handshake trigger types: ACK, NAK, STALL, or NYET.

See "Handshake" on page 827

Special PID ← Trigger type USB
Sets the trigger to one out of four different Special PID trigger types: PREamble, ERR, SPLIT, or PING.

See "Special PID" on page 827

Error condition ← Trigger type USB
Sets the trigger to one out of eight different error condition trigger types: Any error, PID error, CRC5 error, CRC16 error, Bitstuff error, Unexpected PID error, SE1 error, or Glitch error.

See "Error condition" on page 828

Token
Sets the trigger to one out of four different token types:
- "OUT"
- "IN"
- "SOF"
- "SETUP"

If no additional conditions are set, the trigger instant is after the PID.

If the PID error check is selected:
- If an error is found in the PID's complementary form, the trigger is set immediately after the 8th bit of the PID
- If no PID error is found, no trigger is set, even if other selected conditions are met.

Otherwise, the trigger instant is defined by the first occurrence of any of the specified additional conditions.

Remote command:
`TRIGger<m>:USB:TOKen` on page 2160

OUT, IN or SETUP ← Token
For the trigger token types "OUT", "IN" or "SETUP", the following conditions can be set:

You can refine the trigger condition:
- See "PID error " on page 829
- See "End Point" on page 829
- See "Address" on page 829

SOF ← Token
For the trigger token type "SOF", the following conditions can be set:

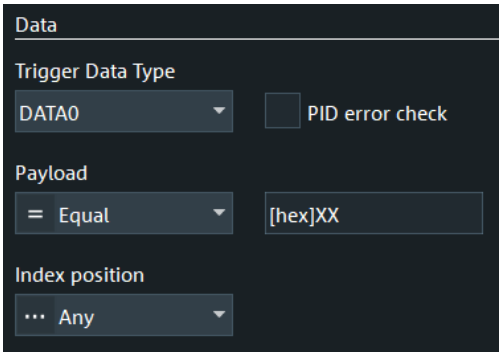
You can refine the trigger condition:
- See "PID error " on page 829
- See "Frame number" on page 830

Data
Sets the trigger to one out of four different data types:
- "DATA0"
- "DATA1"
- "DATA2"
- "MDATA"

If no additional conditions are set, the trigger instant is after the PID.

If the PID error check is selected:
- If an error is found in the PID's complementary form, the trigger is set immediately after the 8th bit of the PID
- If no PID error is found, no trigger is set, even if other selected conditions are met.

Otherwise, the trigger instant is defined by the first occurrence of any of the specified additional conditions.



You can refine the trigger condition:

- See "PID error " on page 829
- See "Payload" on page 829

Remote command:
`TRIGger<m>:USB:DATA` on page 2154

Handshake

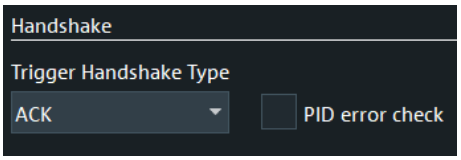
Sets the trigger to one out of four different handshake types:

- "ACK"
- "NAK"
- "STALL"
- "NYET"

If no handshake condition is set, the trigger instant is after the PID.

If the PID error check is selected:

- If an error is found in the PID's complementary form, the trigger is set immediately after the 8th bit of the PID.
- If no PID error is found, no trigger is set.



See "PID error " on page 829

Remote command:
`TRIGger<m>:USB:HAND` on page 2157

Special PID

Sets the trigger to one out of four different "Special PID" types:

- "PREamble"
- "ERR"
- "SPLIT"
- "PING"

If no additional conditions are set, the trigger instant is after the PID.

If the PID error check is selected:

- if an error is found in the PID's complementary form, the trigger is set immediately after the 8th bit of the PID

- if no PID error is found, no trigger is set, even if other selected conditions are met.

Otherwise, the trigger instant is defined by the first occurrence of any of the specified additional conditions.

Remote command:
`TRIGger<m>:USB:SPEC` on page 2159

PREamble or ERR ← Special PID
For the Trigger Special Types "PREamble" and "ERR", the following condition can be set:

See "PID error " on page 829

SPLIT ← Special PID
For the Trigger Special Type "SPLIT", the following conditions can be set:

You can refine the trigger condition:
- See "Address" on page 829
- See "SC" on page 830
- See "Port" on page 830
- See "SEU" on page 831
- See "ET" on page 831
- See "PID error " on page 829

PING ← Special PID
For the Trigger Special Type "PING", the following conditions can be set:

You can refine the trigger condition:
- See "Address" on page 829
- See "End Point" on page 829
- See "PID error " on page 829

Error condition
Sets the trigger in case of one of the following eight different error condition types.

The trigger instant is the first occurrence of the specified error.

"Any error"	Triggers on any of the errors listed below.
"PID error"	Triggers on any packet identifier error.
"CRC5 error"	Triggers on any CRC5 error event
"CRC16 error"	Triggers on any CRC16 error event
"Bitstuff error"	Triggers in the event of an erroneous or missing bit stuffing sequence (see USB standard).
"Unexpected PID error"	Triggers on any illegal PID. This is a PID that is not allowed in USB low speed and USB full speed protocols, especially PID's announcing packets such as SPLIT, DATA2, MDATA, or other noncompliant packets.
"SE1 error"	Triggers on the illegal bus state Single Ended 1 (SE1 = both lines high).
"Glitch error"	Triggers on an error in the bit period (see USB standard for the definition of glitch).

Remote command:
`TRIGger<m>:USB:ERRC` on page 2155

PID error
Defines, whether a packet ID error check is executed or not.

Remote command:
`TRIGger<m>:USB:WPID` on page 2162

End Point
Defines, whether an endpoint check that meets specific conditions is executed or not.

"Condition"	Defining a specific endpoint or an endpoint range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.
"Min"	Specifies an endpoint, or sets the start value of an endpoint range.
"Max"	Sets the end value of an endpoint range if "Condition" (`TRIGger<m>:USB:ECONdition`) is set to `INRange` or `OORange`.

Remote command:
`TRIGger<m>:USB:WEND` on page 2161
`TRIGger<m>:USB:ECONdition` on page 2155
`TRIGger<m>:USB:EMIN` on page 2155
`TRIGger<m>:USB:EMAX` on page 2155

Address
Defines, whether an address check that meets specific conditions is executed or not.

"Condition"	Defining a specific address or an address range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.
"Min"	Specifies an address, or sets the start value of an address range.
"Max"	Sets the end value of an address range if "Condition" (`TRIGger<m>:USB:ACONdition`) is set to `INRange` or `OORange`.

Remote command:
`TRIGger<m>:USB:WADD` on page 2161
`TRIGger<m>:USB:ACONdition` on page 2153
`TRIGger<m>:USB:AMAX` on page 2153
`TRIGger<m>:USB:AMIN` on page 2153

Payload
Defines, whether a payload check that meets specific conditions is executed or not.

"Condition"	Sets the operator "any" or "equal" that allows to trigger for payload data at any position or at a specified position.
"Position"	Available only if "Condition" (`TRIGger<m>:USB:DPOPerator`) is set to *equal*. Specifies the position in which a special data pattern is to be triggered within the payload data packet.
"Data Condition"	Sets the operator ("equal" or "unequal", `TRIGger<m>:USB:DCONdition`) to set a specific payload data pattern.

"Payload pattern" Specifies the payload data pattern that is to be triggered.

Remote command:
`TRIGger<m>:USB:WPAY` on page 2161
`TRIGger<m>:USB:DPOPerator` on page 2154
`TRIGger<m>:USB:DPOSition` on page 2154
`TRIGger<m>:USB:DCONdition` on page 2154
`TRIGger<m>:USB:PATT` on page 2157

Frame number
Defines, whether a frame number check that meets specific conditions is executed or not.

"Condition" Defining a specific frame number or a frame number range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies a frame number, or sets the start value of a frame number range.

"Max" Sets the end value of a frame number range if "Condition" (`TRIGger<m>:USB:FCONdition`) is set to `INRange` or `OORange`.

Remote command:
`TRIGger<m>:USB:WFRN` on page 2161
`TRIGger<m>:USB:FCONdition` on page 2156
`TRIGger<m>:USB:FMIN` on page 2156
`TRIGger<m>:USB:FMAX` on page 2157

SC
Defines, whether a Start / Complete SPLIT transaction check is executed or not.

Remote command:
`TRIGger<m>:USB:WSTC` on page 2162
`TRIGger<m>:USB:STCO` on page 2159

Port
Defines, whether a port check that meets specific conditions is executed or not.

"Condition" Defining a specific port number or a port number range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies a port number, or sets the start value of a port number range.

"Max" Sets the end value of a port number range if "Condition" (`TRIGger<m>:USB:PCONdition`) is set to `INRange` or `OORange`.

Remote command:
`TRIGger<m>:USB:WPOR` on page 2162
`TRIGger<m>:USB:PCONdition` on page 2157

TRIGger<m>:USB:PMIN on page 2158
TRIGger<m>:USB:PMAX on page 2158

SEU
Defines, whether an SEU check that meets specific conditions is executed or not. S and E represent the Start and End of a start-split transaction, U represents the reserved/Unused bit of a complete-split transaction. Permissible binary SEU values are 00, 01, 10, and 11.

"Condition" Defining a specific SEU value or an SEU value range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies an SEU value, or sets the start value of an SEU value range.

"Max" Sets the end value of an SEU value range if "Condition" (TRIGger<m>:USB:SCONdition) is set to INRange or OORange.

Remote command:
TRIGger<m>:USB:WSEU on page 2162
TRIGger<m>:USB:SCONdition on page 2158
TRIGger<m>:USB:SMIN on page 2158
TRIGger<m>:USB:SMAX on page 2159

ET
Defines, whether an Endpoint Type (ET) check that meets specific conditions is executed or not.

"Condition" Defining a specific endpoint type or an ET range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies an ET, or sets the start value of an ET range.

"Max" Sets the end value of an ET range if "Condition" (TRIGger<m>:USB:TCONdition) is set to INRange or OORange.

Remote command:
TRIGger<m>:USB:WETCheck on page 2161
TRIGger<m>:USB:TCONdition on page 2160
TRIGger<m>:USB:TMIN on page 2160
TRIGger<m>:USB:TMAX on page 2160

13.20.3.2 Triggering on USB 2.0

Prerequisite: A bus is configured for the USB 2.0 signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Press "Setup trigger".

3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the serial bus that is set to USB 2.0.
5. Tap "Trigger Type USB" and select the trigger type to be used for USB protocol analysis.
   Available trigger types depend on the USB 2.0 protocol type that is activated in "Protocol type" on page 819.
6. Depending on the selected USB 2.0 protocol type and trigger type, more setup conditions have to be specified.

For information on how to proceed with the configuration settings, see Chapter 13.20.3.1, "USB 2.0 trigger settings", on page 824.

13.20.4 USB 2.0 decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Serial Bus" dialog > "Setup" tab, enable "Decode".
2. In the "Serial Bus" dialog > "Display" tab, select additional result display settings: "Show table" and "Show binary".
3. If required, enable "Zoom coupling"

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

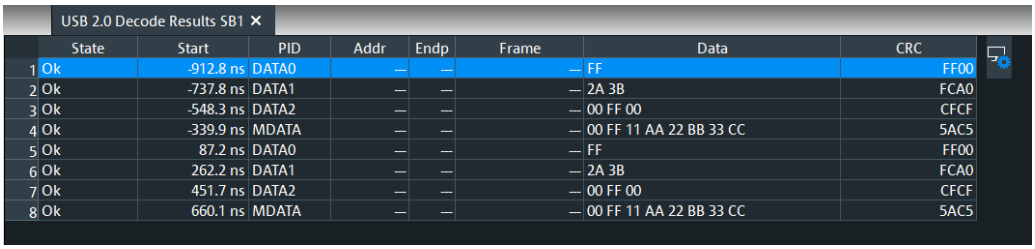
The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.


USB 2.0 Decode Results SB1

	State	Start	PID	Addr	Endp	Frame	Data	CRC
1	Ok	-912.8 ns	DATA0	—	—	—	FF	FF00
2	Ok	-737.8 ns	DATA1	—	—	—	2A 3B	FCA0
3	Ok	-548.3 ns	DATA2	—	—	—	00 FF 00	CFCF
4	Ok	-339.9 ns	MDATA	—	—	—	00 FF 11 AA 22 BB 33 CC	5AC5
5	Ok	87.2 ns	DATA0	—	—	—	FF	FF00
6	Ok	262.2 ns	DATA1	—	—	—	2A 3B	FCA0
7	Ok	451.7 ns	DATA2	—	—	—	00 FF 00	CFCF
8	Ok	660.1 ns	MDATA	—	—	—	00 FF 11 AA 22 BB 33 CC	5AC5

Table 13-34: Content of the Decode results table

Column	Description
State	Overall state of the packet: either OK or the relevant error condition (CRC, glitching, ...)
Start	Start time of the packet.
PID	PID type (OUT, IN, DATA0, ...)
Addr	Address of the recipient
Endp	Endpoint of the recipient
Frame	Frame number (in SOF packet)
Data	Values of the payload data bytes. The data format is selected below the table.
CRC	Either CRC5 or CRC16 (data packet PID)

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands to retrieve decode results are described in Chapter 24.17.21.3, "Decode results", on page 2162.

13.20.5 Search on decoded USB 2.0 data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

Before you can start the search, you have to configure the bus correctly and acquire decoded data.

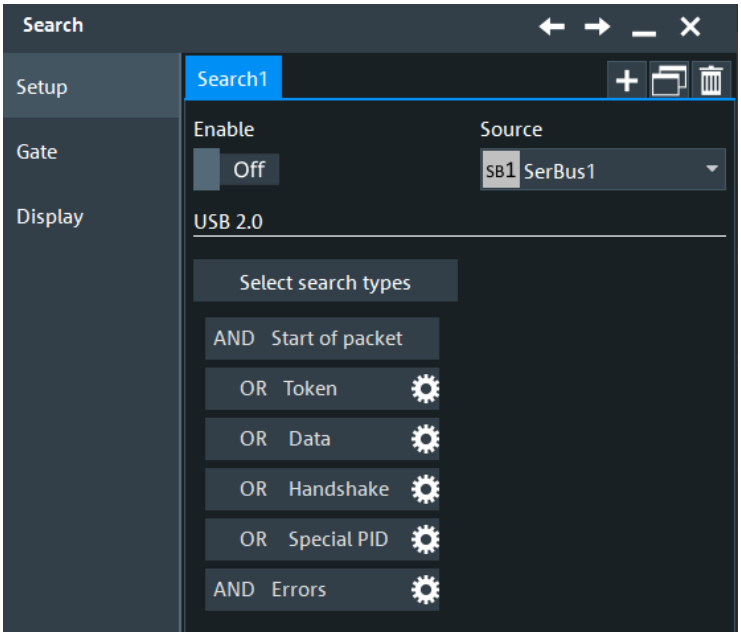
To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 11, "Search functions", on page 419.

If you need information on how to get started with searching USB 2.0 data, see Chapter 13.20.5.3, "Searching USB data", on page 839. Otherwise proceed with the USB 2.0 search setup.

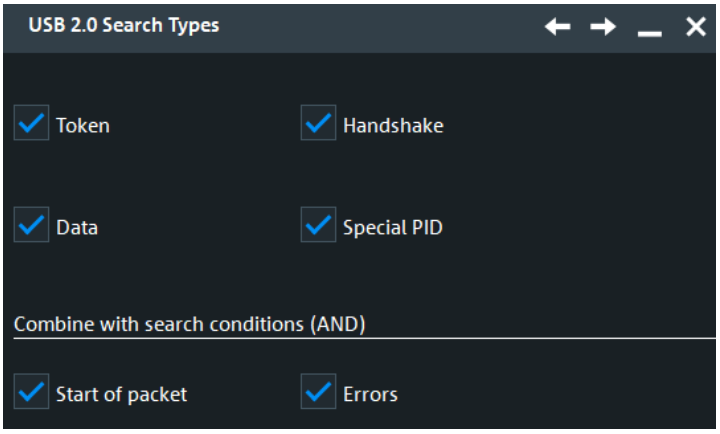
13.20.5.1 USB 2.0 search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"



Search types

Use the "Search Types" dialog to define the event types to be searched for.



Individual search parameters (which do not depend on the USB protocol type and trigger settings), can be specified in the tabs below the "Search criteria" dialog.

Remote command:

`SEARch:TRIGger:USB:SSOP` on page 2177
`SEARch:TRIGger:USB:STOKen` on page 2179
`SEARch:TRIGger:USB:SDATa` on page 2177
`SEARch:TRIGger:USB:SHANdshake` on page 2177
`SEARch:TRIGger:USB:SSPE` on page 2178
`SEARch:TRIGger:USB:SERRor` on page 2177

SOP

Searches for any start of packet. There are no additional parameters to be defined.

Token
Searches for four different token types: "OUT", "IN", "SOF", or "SETUP", and "Any" token.

For "Any" token, there are no additional parameters to be defined.

For "OUT", "IN", or "SETUP" tokens, additional search parameters are "Address check" and "End Point check".

You can refine the search condition:
- See "Address check" on page 836
- See "End Point check" on page 836

For "SOF" tokens, the additional search parameter is "Frame number check".

You can refine the search condition:
- See "Frame number check" on page 837

Remote command:
`SEARch:TRIGger:USB:TOKen` on page 2180

Data
Searches for data packets of the following types: DATA0, DATA1, DATA2, or MDATA, and "Any" data packet.

To search for payload in any data packet type, a data pattern and optionally a packet position have to be specified.

You can refine the search condition:
- See "Payload check" on page 837

Remote command:
`SEARch:TRIGger:USB:DATA` on page 2171

Handshake
Searches for four different handshake packet types: "ACK", "NAK", "STALL", or "NYET", and "Any" handshake packet. There are no additional parameters to be defined.

Remote command:
`SEARch:TRIGger:USB:HAND` on page 2174

Special PID
Searches for four different special packet identifier types: "PREamble", "ERR", "SPLIT", or "PING", and "Any" special PID.

For "Any", "PREamble", or "ERR", there are no additional parameters to be defined.

For "SPLIT", additional search parameters are "Address check", "SC check", "Port check", "SEU check", and "ET check".

You can refine the search condition:
- See "Address check" on page 836
- See "SC check" on page 838
- See "Port check" on page 838
- See "SEU check" on page 838
- See "ET check" on page 838

For "PING", additional search parameters are "Address check" and "End Point check".

You can refine the search condition:

- See "Address check" on page 836
- See "End Point check" on page 836

Remote command:
`SEARch:TRIGger:USB:SPEC` on page 2178

Error condition
Searches for the following error conditions:

"PID Error" Searches for packet identifier errors.

Remote command:
`SEARch:TRIGger:USB:PIDerror` on page 2175

"CRC5 Error" Searches for any CRC5 error event.

Remote command:
`SEARch:TRIGger:USB:CRC5error` on page 2170

"CRC16 Error" Searches for any CRC16 error event.

Remote command:
`SEARch:TRIGger:USB:CRC16error` on page 2170

"Bitstuff Error" Searches for bitstuff errors, thus an erroneous or missing bit stuffing sequence (see USB standard).

Remote command:
`SEARch:TRIGger:USB:BITSterror` on page 2170

"Glitch Error" Searches for glitch errors (errors in the bit period, see USB standard for the definition of glitch).

Remote command:
`SEARch:TRIGger:USB:GLITcherror` on page 2174

Address check
Enables the search for an address that meets specific conditions.

"Condition" Defining a specific address or an address range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies an address, or sets the start value of an address range.

"Max" Sets the end value of an address range if "Condition" (`SEARch:TRIGger:USB:ACONdition`) is set to `INRange` or `OORange`.

Remote command:
`SEARch:TRIGger:USB:WADD` on page 2180
`SEARch:TRIGger:USB:ACONdition` on page 2169
`SEARch:TRIGger:USB:AMIN` on page 2169
`SEARch:TRIGger:USB:AMAX` on page 2170

End Point check
Enables the search for an endpoint that meets specific conditions.

"Condition" Defining a specific endpoint or an endpoint range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies an endpoint, or sets the start value of an endpoint range.

"Max" Sets the end value of an endpoint range if "Condition" (`SEARch:TRIGger:USB:ECONdition`) is set to `INRange` or `OORange`.

Remote command:
`SEARch:TRIGger:USB:WEND` on page 2181
`SEARch:TRIGger:USB:ECONdition` on page 2172
`SEARch:TRIGger:USB:EMIN` on page 2173
`SEARch:TRIGger:USB:EMAX` on page 2173

Frame number check
Enables the search for a frame number that meets specific conditions.

"Condition" Defining a specific frame number or a frame number range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies a frame number, or sets the start value of a frame number range.

"Max" Sets the end value of a frame number range if "Condition" (`SEARch:TRIGger:USB:FCONdition`) is set to `INRange` or `OORange`.

Remote command:
`SEARch:TRIGger:USB:WFRN` on page 2181
`SEARch:TRIGger:USB:FCONdition` on page 2173
`SEARch:TRIGger:USB:FMIN` on page 2173
`SEARch:TRIGger:USB:FMAX` on page 2174

Payload check
Enables the search for a payload data pattern that meets specific conditions.

"Condition" Sets the operator "any" or "equal" that allows to search for payload data at any position or at a specified position.

"Position" Available only if "Condition" (`SEARch:TRIGger:USB:DPOPerator`) is set to *equal*. Specifies the position in which a special data pattern is to be searched within the payload data packet.

"Data Condition" Sets the operator ("equal" or "unequal", `SEARch:TRIGger:USB:DCONdition`) to set a specific payload data pattern.

"Payload pattern" Specifies the payload data pattern that is to be searched.

Remote command:
`SEARch:TRIGger:USB:WPAY` on page 2181
`SEARch:TRIGger:USB:DPOPerator` on page 2171
`SEARch:TRIGger:USB:DPOSition` on page 2172
`SEARch:TRIGger:USB:DCONdition` on page 2171
`SEARch:TRIGger:USB:PATT` on page 2174

SC check
Searches for the selected Start (0) or Complete (1) SPLIT transaction endpoint, or X (do not care).

Remote command:
`SEARch:TRIGger:USB:WSTC` on page 2182
`SEARch:TRIGger:USB:STCO` on page 2178

Port check
Enables the search for a port that meets specific conditions.

"Condition" Defining a specific port number or a port number range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies a port number, or sets the start value of a port number range.

"Max" Sets the end value of a port number range if "Condition" (`SEARch:TRIGger:USB:PCONdition`) is set to `INRange` or `OORange`.

Remote command:
`SEARch:TRIGger:USB:WPOR` on page 2182
`SEARch:TRIGger:USB:PCONdition` on page 2175
`SEARch:TRIGger:USB:PMIN` on page 2175
`SEARch:TRIGger:USB:PMAX` on page 2175

SEU check
Enables the search for an SEU that meets specific conditions. (For SEU, see "SEU" on page 831.)

"Condition" Defining a specific SEU value or an SEU value range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies an SEU value, or sets the start value of an SEU value range.

"Max" Sets the end value of an SEU value range if "Condition" (`SEARch:TRIGger:USB:SCONdition`) is set to `INRange` or `OORange`.

Remote command:
`SEARch:TRIGger:USB:WSEU` on page 2182
`SEARch:TRIGger:USB:SCONdition` on page 2176
`SEARch:TRIGger:USB:SMIN` on page 2176
`SEARch:TRIGger:USB:SMAX` on page 2176

ET check
Enables the search for an Endpoint Type (ET) that meets specific conditions.

"Condition" Defining a specific endpoint type or an ET range requires to set the operator to one of the following: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.

"Min" Specifies an ET, or sets the start value of an ET range.

"Max" Sets the end value of an ET range if "Condition" (`SEARch:TRIGger:USB:TCONdition`) is set to `INRange` or `OORange`.

Remote command:
`SEARch:TRIGger:USB:WETCheck` on page 2181
`SEARch:TRIGger:USB:TCONdition` on page 2179
`SEARch:TRIGger:USB:TMIN` on page 2179
`SEARch:TRIGger:USB:TMAX` on page 2180

13.20.5.2 USB 2.0 search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands:

- `SEARch:RESult:USB:PCOunt?` on page 2187
- `SEARch:RESult:USB:PACKet<m>:STATus?` on page 2187
- `SEARch:RESult:USB:PACKet<m>:STARt?` on page 2186
- `SEARch:RESult:USB:PACKet<m>:STOP?` on page 2187
- `SEARch:RESult:USB:PACKet<m>:ADDRess?` on page 2183
- `SEARch:RESult:USB:PACKet<m>:DATA?` on page 2184
- `SEARch:RESult:USB:PACKet<m>:CRC?` on page 2183
- `SEARch:RESult:USB:PACKet<m>:ENDPoint?` on page 2184
- `SEARch:RESult:USB:PACKet<m>:ET?` on page 2184
- `SEARch:RESult:USB:PACKet<m>:FRAMe?` on page 2184
- `SEARch:RESult:USB:PACKet<m>:PID?` on page 2185
- `SEARch:RESult:USB:PACKet<m>:PORT?` on page 2185
- `SEARch:RESult:USB:PACKet<m>:SC?` on page 2186
- `SEARch:RESult:USB:PACKet<m>:SEU?` on page 2186

13.20.5.3 Searching USB data

Prerequisite: A serial bus is configured for the USB signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Tap "Setup search".

3. If the dialog box does not contain a search entry, tap the "Add" icon to create one, as described in "To create a user-defined search" on page 434.
4. Tap "Source" and select the serial bus that is set to USB 2.0 (e.g. "SerBus1", unless already selected).
5. Specify search criteria according to Chapter 13.20.5.1, "USB 2.0 search setup", on page 833.
6. To acquire a waveform, press [Single].

   The R&S RTO6 performs a USB 2.0 decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).
7. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog.

   The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.21 USB 3.1 (option R&S RTO6-K570)

Option R&S RTO6-K570 analyzes Universal Serial Bus (USB) signals encoded by the USB 3.1 Gen1 coding standard. To use the full functionality of the option, you need an instrument that covers the frequency range up to 5 GHz, because the USB 3.1 Gen 1 protocol is defined for data rates up to 5 GHz.

For analysis, USB 3.1 encoded signals can be triggered, decoded and searched.

This chapter describes:

- The USB 3.1 protocol....840
- USB 3.1 configuration....848
- USB 3.1 trigger....853
- USB 3.1 decode results....856
- Search on decoded USB 3.1 data....858

13.21.1 The USB 3.1 protocol

The USB 3.1 protocol standard was published on July 26th, 2013, strongly enhancing the performance of the USB 2.0 protocol. Compared to USB 2.0, USB 3.1 defines a new physical layout, better power distribution and higher data rates.

The USB 3.1 standard features two speed levels (generations):

- **USB 3.1 Gen 1** for 5 Gbps (also labeled "SuperSpeed" or "SS"), encompassing and superseding USB 3.0 (published on November 12th, 2008)
- **USB 3.1 Gen 2** for 10 Gbps (also labeled "SuperSpeedPlus" or "SSP")

While USB 3.1 Gen 2 is not yet very common, you can use a fast oscilloscope (minimum 5 GHz) with option **R&S RTO6-K570** to analyze signals encoded according to the **USB 3.1 Gen 1** protocol.

NOTICE

Oscilloscope must range up to 5 GHz, at least

As the USB 3.1 Gen 1 protocol is defined for data rates up to 5 GHz, option R&S RTO6-K570 requires a fast oscilloscope that covers this frequency range.

USB 3.1 supersedes the USB 3.0 standard but uses, for example, the USB 3.0 Standard-A connector design with 9 pins. Hence, USB 3.1 cables and connectors contain 5 additional wires and pins compared to USB 2.0.

Figure 13-45: USB 3.1 cable cross-section and Standard-A plug - backward compatible with USB 2.0

A = Shield (braid) / connector shell
B = USB 2.0 unshielded twisted pair
C = USB 3.0 / USB 3.1 shielded twisted pair
D = USB 3.0 / USB 3.1 shielded twisted pair
1 = Voltage bus V_{CC} power supply pin, +5 V
2 = Differential data signal D-
3 = Differential data signal D+
4 = Ground pin for power return
5 = SuperSpeed receiver differential pair Rx-
6 = SuperSpeed receiver differential pair Rx+
7 = Ground drain for signal return
8 = SuperSpeed transmitter differential pair Tx-
9 = SuperSpeed transmitter differential pair Tx+
* = Connector images courtesy of Wikipedia authors smial and Unconventional2

Table 13-35: Backward compatibility of USB 3.1 connectors

Connector type	USB 3.1	Compatibility	USB 2.0
Standard-A	plug	... is backward compatible with ...	receptacle
	receptacle	... is backward compatible with ...	plug
Standard-B	plug	... is **not** compatible with ...	receptacle
	receptacle	... is backward compatible with ...	plug
Micro-USB	plug	... is **not** compatible with ...	receptacle
	receptacle	... is backward compatible with ...	plug

To achieve the data throughput of USB 3.1, all involved components (host, cable, device and optional hub) must comply with USB 3.1 specifications. If any component only complies, for example, with USB 2.0 Hi-Speed specifications, the setup works, but limited to Hi-Speed USB data rates.

USB 3.1 also specifies a new connector format, called Type-C, with a reversible plug. This small and durable connector is, however, not mechanically compatible with USB 2.0 connectors.

USB 3.1 Type-C connector

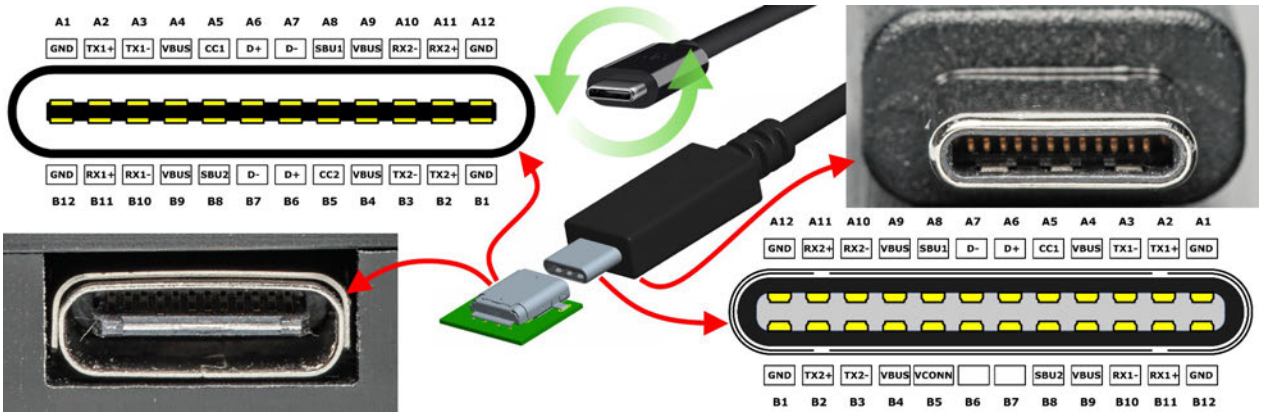


Figure 13-46: USB 3.1 Type-C receptacle and reversible plug

A1 to A12 = Twelve USB Type-C pins
B1 to B12 = The same USB Type-C pins as A1 to A 12, but in reverse order
* = Images courtesy of Heise Medien, Acon, phoneArena and Wikipedia author Chindi.ap

Specific protocol features

Key features of USB 3.1 Gen 1 that are addressed by option R&S RTO6-K570 include:

- Since 5 Gbps is too fast for serial pattern triggering, the option uses a search-based software trigger
- The protocol uses a non-return-to-zero (NRZ) unclocked signal with **8b/10b encoding** (while USB 3.1 Gen 2 uses 128b/132b encoding)
- The signal is scrambled with a 32-bit minimum correlation sequence
- COMMA (K28.5), which occurs only during the reset phase, is the only unique bit sequence
  - COMMA is used for reset of the scrambling sequence, which has 32-bit minimum correlation
  - COMMA and packet framing are used for bit alignment
- The protocol uses a set of different frame types that are all covered by individual trigger and search dialogues of option R&S RTO6-K570

Frame types

All frame types are listed below. The frame types are sorted in an almost alphabetical order, as in the oscilloscope's graphical user interface (GUI).

Each item in this list contains:

- the SCPI name of the frame
  (hence the name of the frame according to the related `remote control command (SCPI)`)
- the GUI name of the frame, which is sometimes identical with the SCPI name (the GUI names are highlighted in **bold** font and linked to one of Table 13-36 / Table 13-37 / Table 13-38)
- a brief description of the frame type

List of frame types

- ACK - **ACK** = Handshake packet indicating a positive acknowledgment.
- BCNT - **BCNT** = BERT count, with BERT = Bit Error Rate Test. BCNT is a K28.3 sequence followed by the error count symbol EC. See also BERC.
- BDAT - **BDAT** = BERT data, ordered set of a BERT command. This data sequence (output of the scrambler) is used for BER testing and is equivalent to the logical idle sequence, consisting of scrambled 0.
- BERC - **BERC** = BERT error count, ordered set of a BERT command that does not increment the error count register. BERC is not looped back to the receiver but replaced with BCNT.
- BRST - **BRST** = BERT reset, ordered set of a BERT command. Any time a BRST is received, the error count register EC is set to 0 and the linear feedback shift register (LFSR) for scrambling is set to 0FFFFh. Any number of consecutive BRST ordered sets may be received.
- BIAM - **Bus Interval Adjustment** = The bus interval is the period that establishes the integral boundary of service intervals. It is equivalent to the Microframe interval (THSFRAM) defined in the USB 2.0 specification.
- CP0 - compliance data pattern. Runs in scrambled mode.
- CP1/CP2 - compliance data pattern. Run in unscrambled mode.
- CP3/CP5or6 - compliance data pattern. Run in both scrambled and unscrambled mode.
- DPH - **DPH** = Data Packet Header, containing a type field, the data packet's address, route string, length, and other information about the packet, including a 2-byte checksum (CRC-16).
- DPH n-def - = non-deferred Data Packet Header, containing a type field, the data packet's address, route string, length field replica, and other information about the packet, including a 2-byte checksum (CRC-16). Valid only for USB 3.1 Gen 2.
- DPP - **Data Packet Payload** = The packet's payload data and a 32-bit CRC.
- DPPA - **Data Packet Payload Abort** = Frame ordered set used to abort a data packet payload.
- ERDY - **ERDY** = The Endpoint Ready notification is a handshake packet indicating a positive acknowledgment.
- FWAKe - **Function Wake** = The function wake device notification is sent from a device (that is enabled for remote wakeup) to the host, if it wants to exit from "device suspend" or "function suspend".

- HRR - **Host Role Request** = This notification type is reserved for being issued by a SuperSpeed On-the-Go (SS-OTG) device, if its Role Swap Protocol (RSP) capability has been enabled.
- IDLE - **IDLE** = Logical Idle is a period of one or more symbol periods when no information (packets or link commands) is transferred on the link. The Idle Symbol (IS) is transmitted by a port at any time in the active (U0) state meeting the logical idle definition.
- ITP - **ITP** = Isochronous Timestamp Packet, sent periodically by a host to inform devices on the USB of the current bus time. ITPs are multicast on all active links.
- LTM - **Latency Tolerance** = Latency Tolerance Messaging is an optional normative USB power management feature that utilizes reported BELT (Best Effort Latency Tolerance) values to enable more power efficient platform operation.
- LC - **Link Command** = An eight-symbol sequence used for link-level flow control, retries, power management and device removal.
- LDM - **Link Delay Measurement** = A protocol for determining propagation delays through the USB topology with a high degree of accuracy (sometimes called "Precision Time Measurement").
- NRDY - **NRDY** = The Not Ready response is a handshake packet indicating a negative acknowledgment.
- PING - **PING** = This transaction packet can only be sent by the host, to transition all links, which are in the path to a device, back to the active (U0) state prior to initiating an isochronous transfer.
- PNGR - **PING_RESPONSE** = This transaction packet is sent by a device in response for each PING received from the host.
- PCAP - **Port Capability** = This link management packet describes each port's link capabilities and is sent by both link partners after the successful completion of training and link initialization.
- PCNF - **Port Configuration** = All Enhanced SuperSpeed ports that support downstream port capability can send this link management packet (LMP), which is similar to Port Capability. If a port, which was to be configured in the upstream facing mode, does not receive this LMP within a defined time after link initialization, then the upstream port shall transition to "eSS.Disabled" and a peripheral device shall try to connect at the other speeds this device supports.
- PCNR - **Port Configuration Response** = This link management packet is sent by the upstream port in response to a Port Configuration and indicates its acceptance or rejection.
- SDS - **Start of Data Stream** marks the start of an SDS ordered set. Valid only for USB 3.1 Gen 2.
- SKIP - **SKIP** = SKP compensates for different bit rates between two communicating ports. SKPs may be dynamically inserted or removed from the data stream. SKPEND marks the boundary between SKP symbols and the remainder of the SKP ordered set. (For SuperSpeedPlus operation, unscrambled.)
- STALL - **STALL** = This transaction packet can only be sent by an endpoint on the device. It is used to inform the host that the endpoint is halted or that a control transfer is invalid.

- STATUS - **STATUS** = This transaction packet can only be sent by the host to a control endpoint. It is used to inform a control endpoint that the host has initiated the Status stage of a control transfer.
- SLF - **Set Link Function** = This link management packet is used to configure functionality that can be changed without leaving the active (U0) state.
- SSPD - **Sublink Speed** = Speed of a sublink, which is the collection of receive or transmit lanes between a downstream facing port (DFP) and an upstream facing port (UFP).
- TS1/TS1A/TS1B - **TS1** = Training sequences are ordered sets for initializing bit and symbol alignment and receiver equalization. Examples are TS1, TS2 and TSEQ.
- TS2 - **TS2** = See TS1
- TSEQ - **TSEQ** = See TS1
- U2IT - **U2 Inactivity Timeout** = This link management packet is used to define the timeout from U1 to U2.
- VDT - **Vendor Device Test** = Use of this link management packet is intended for vendor-specific device testing and shall not be used during normal operation of the link.

Other types

- ERRor - **Error** = This is not a frame type, but it is included as an artificial entry in the search and navigate dialog to allow selecting and activating the error search. If the host error bit is set to "1", this state indicates that the host was unable to accept the data. The reason can be a CRC error, length error, value out of range, packet incomplete or unknown error.
- UDEF - **Unknown** = Undefined data cannot be identified as any specific USB 3.1 frame, because the measured signal does not match with the specifications of any defined frame. Therefore, in this case the field is represented by a "?". You cannot search for "Unknown" or trigger on it, but these types are returned, if an unexpected (sub)type has been found. If some rough type identification is possible, the unknown data may fit one of the following descriptions of unknown types.
- UDVN - **Unknown Device Notification** = A Device Notification transaction packet can only be sent by a device, to inform the host of an asynchronous change in a device or interface state (e.g., to identify the function within a device that caused the device to perform a remote wake operation).
- UHP - **Unknown Header Packet** = An undefined header packet that starts with HPSTART, but then the type is invalid.
- ULMP - **Unknown LMP** = A Link Management Packet is a type of header packet primarily used to manage links by communicating information between links partners. LMPs only travel between pairs of directly connected ports.
- UTP - **Unknown TP** = A Transaction Packet is a type of header packet used to communicate information between a device and the host. TPs traverse all the links, directly connecting the host to a device. They have no payload data, instead they are used, e.g., to control the flow of data packets or configure devices and hubs.

The frames listed above consist of individual sets of fields. Some frames only contain one field, others are much longer: up to 29 fields (in case of the "ACK" frame).

We try to represent the structure of these frames. However, one single table that contains all frames and all fields would be very large: 30 by 40 cells. We have therefore split up the full table into the following four smaller tables:

- Table 13-36 contains the **short** frames that end with the "Link Control Word". (This table uses abbreviation, see below.)
- Table 13-37 contains the **long** frames that end with the "Link Control Word". (This table uses abbreviation, see below.)
- Table 13-38 contains the frames that do not end with the "Link Control Word". These frames have no additional fields other than those in the table.
- Table 13-39 contains the types that are no specified USB 3.1 frames.

The frames in both Table 13-36 and Table 13-37 are abbreviated. For the full set of fields, you must add to each frame the 2 fields "HPSTART" and "Type" in the beginning of the frame. Also add a "CRC" field and the 6 fields of the "Link Control Word" at the end of each frame. In Table 13-39, only the "Link Control Word" is abbreviated.

The 2-byte **"Link Control Word"** is used for both link level and end-to-end flow control. The fields in the Link Control Word are detailled in Table 13-37.

Table 13-36: Short frames that start with fields HPSTART and Type and end with CRC and the Link Control Word

Frame	Field3	Field4	Field5	Field6	Field7	Field8	Fld.9	Field10	Fld.11
Bus Interval Adjustment	Rsvd	Device Address	SubType	NotType	Rsvd	BusIntAdj	Rsvd		
Function Wake	Rsvd	Device Address	SubType	NotType	Interface	Rsvd	Rsvd		
Host Role Request	Rsvd	Device Address	SubType	NotType	RSP	Rsvd	Rsvd		
ITP	Counter	Delta	Adj Ctrl	Corr	Rsvd	Rsvd			
Latency Tolerance	Rsvd	Device Address	SubType	NotType	BELT	Rsvd	Rsvd		
Link Delay Measurement	SubType	LDM Type	LDMS	Rsvd	Response Delay	Rsvd	Rsvd		
NRDY	Rsvd	Device Address	SubType	Rsvd	D	EPTNum	Rsvd	StreamId	Rsvd
PING	Route String	Device Address	SubType	Rsvd	D	EPTNum	Rsvd	Rsvd	
PING_RESPONSE	Rsvd	Device Address	SubType	Rsvd	D	EPTNum	Rsvd	Rsvd	
Port Configuration	SubType	Link Speed	Rsvd	Rsvd	Rsvd				
Port Configuration Response	SubType	Response Code	Rsvd	Rsvd	Rsvd				
STALL	Rsvd	Device Address	SubType	Rsvd	D	EPTNum	Rsvd	Rsvd	
Set Link Function	SubType	SLF	Rsvd	Rsvd	Rsvd				
U2 Inactivity Timeout	SubType	U2 IT	Rsvd	Rsvd	Rsvd				
Vendor Device Test	SubType	VDT	Rsvd	Vendor Def	... for fields no. 1, 2 and n+x see Table 13-37				

Table 13-37: Long frames that start with fields HPSTART and Type and end with CRC and the Link Control Word

Frame	Fld.3	Field4	Fld.5	Field6	Fld.7	Fld.8	Field9	F.10	Fld.11	Fld.12	Fld.13	F.14	F.15	Fld.16	F.17	F.18	F.19	F.20	F.21	F.22
ACK	Route String	Device Address	Sub-Type	Rsvd	rty	D	EPT-Num	TT	HE	NumP	SeqNum	Rsvd	TPF	StreamId	Rsvd	SSI	WPA	DBI	PP	NBI
DPH	Route String	Device Address	Seq-Num	Rsvd	EOB	D	EPT-Num	TT	S	Length	StreamId	Rsvd	SSI	WPA	DBI	PP	NBI			
ERDY	Rsvd	Device Address	Sub-Type	Rsvd	D	EPT-Num	Rsvd	Num P	Rsvd	StreamId	Rsvd									
Port Capa-bility	Sub-Type	Link Speed	Rsvd	Num HP Buf	Rsvd	Dir	OTG	Rsvd	Tie-breaker	Rsvd	Rsvd									
STA-TUS	Route String	Device Address	Sub-Type	Rsvd	D	EPT-Num	Rsvd	Rsvd	PP	Rsvd										
Sublink Speed	Rsvd	Device Address	Sub-Type	Not-Type	Rsvd	TPF	Rsvd	LSE	ST	Rsvd	Lanes	LP	LSM							
Ahead of **Field 3**, add these two fields (**Field1** + **Field2**):							**Field1**		**Field2**		**Field3**		...							
							HPSTART		Type		see above		...							
After the **last field (n)**, add CRC + the **"Link Control Word"**:							**Field n+1**		**Field n+2**		**Field n+3**		**Field n+4**		**Field n+5**		**Field n+6**		**Field n+7**	
							CRC		**Hdr#**		**Rsvd**		**Hub#**		**DLY**		**DFR**		**CRC**	

Table 13-38: Frames without additional fields other than in this table

Frame	Field1	Field2	Field3	Field4	Field5	Field6	F.7	Fld.8	Field9
BCNT	BERC	EC							
BDAT	BDAT								
BERC	BERC								
BRST	COM	BRST							
Data Packet Payload	HPSTART	Data	CRC	DPPEND					
Data Packet Payload Abort	HPSTART	Data	CRC	DPPABORT					
IDLE	Idle								
Link Command	LCSTART	SubType	CRC	SubType	CRC				
SKIP	SKP								
TS1	COM	Rsvd	Train	Rsvd	Loop	Scramble	LL	Rsvd	TS1ID
TS2	COM	Rsvd	Train	Rsvd	Loop	Scramble	LL	Rsvd	TS2ID
TSEQ	COM	Symbol							

And finally the remaining types that are no specified USB 3.1 frames:

Table 13-39: Other types

Type	Field1	Field2	Field3	Field4	Field5	Field6	Field7	Field8	Field9
Error	CRC Error	Length Error	Unknown	Value out of range	Packet Incomplete				
Unknown	?								
Unknown Device Notification	HPSTART	Type	Rsvd	Device Address	SubType	NotType	?	CRC	**LCW**
Unknown Header Packet	HPSTART	Type	?	?	CRC	**LCW**			
Unknown LMP	HPSTART	Type	SubType	?	?	CRC	**LCW**		
Unknown TP	HPSTART	Type	Rsvd	Device Address	SubType	?	?	CRC	**LCW**
LCW = Link Control Word, see Table 13-37, occupies the last 6 fields.									

More information on the USB 3.1 protocol, including all specifications down to the field contents, is available in the "Universal Serial Bus 3.1 Specification" documentation. Refer to the online resources at www.usb.org.

13.21.2 USB 3.1 configuration

If you need information on how to get started with configuring the USB setup, see Chapter 13.21.2.3, "Configuring USB 3.1 signals", on page 852. Otherwise proceed with the configuration settings.

13.21.2.1 USB 3.1 configuration settings

Access: [Bus] > "USB 3.1" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.

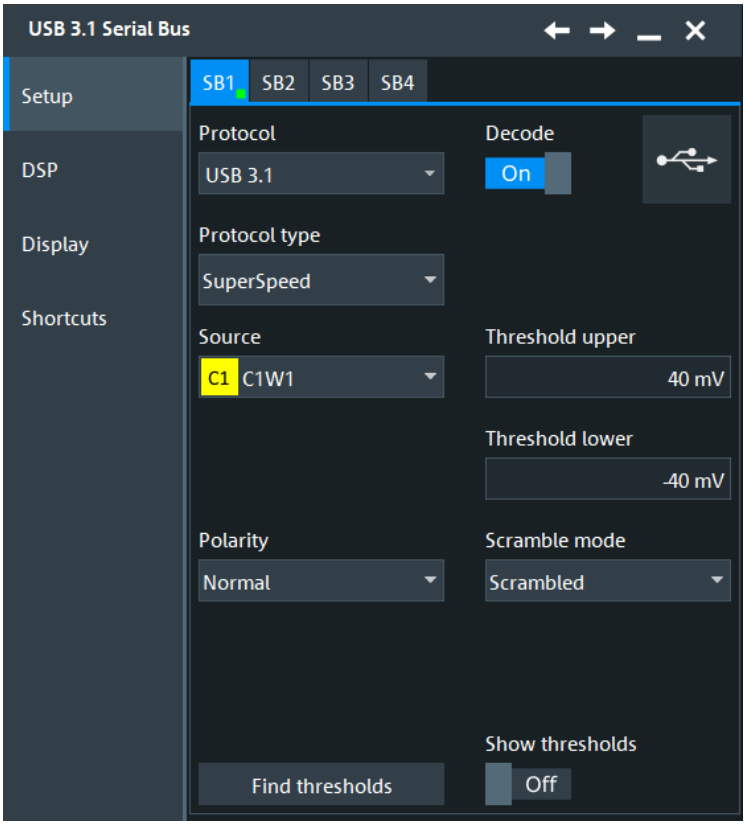


Figure 13-47: Serial bus protocol configuration dialog

See also: Chapter 13.1.1, "Setup - general settings", on page 485

Protocol type
Selects the USB 3.1 protocol type.

"SuperSpeed"
USB 3.1 Gen 1 with 5 Gbit/s data rate.

Remote command:
`BUS<m>:USBThree:TYPE` on page 2189

Source
Defines the source for the data signal. You can select analog channels "C1" to "C4", and also active math and reference waveforms as source. If you want to trigger on the serial bus, analog channel sources are required.

Remote command:
`BUS<m>:USBThree:SOURce` on page 2188

Polarity
Defines the signal polarity settings. Permitted selections are "Normal" and "Inverted".

Remote command:
`BUS<m>:USBThree:POLarity` on page 2188

Thresholds
Sets the threshold values for digitization of the analog signals.

The USB 3.1 standard uses 2-state signals with an upper and a lower voltage level. Permitted thresholds values are in the range of -2 V to +2 V, the default is -40 mV to +40 mV.

If the signal value on the line is higher than the upper threshold, the signal state is considered high. Otherwise, if the signal value is below the lower threshold, the signal state is considered low. The two threshold levels allow configuring a hysteresis setting.

There are two ways to set the thresholds:

"Threshold"	Enter the upper and lower values directly into the fields.
"Find thresholds"	Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
`BUS<m>:USBThree:THRHigh` on page 2189
`BUS<m>:USBThree:THRLow` on page 2189
`BUS<m>:SETReflevels` on page 1671

Scramble Mode
Selects if the data pattern is scrambled or unscrambled. Scrambling as specified in the standard minimizes EMI emissions, the unscrambled mode can simplify testing and debugging.

Remote command:
`BUS<m>:USBThree:SCRMode` on page 2189

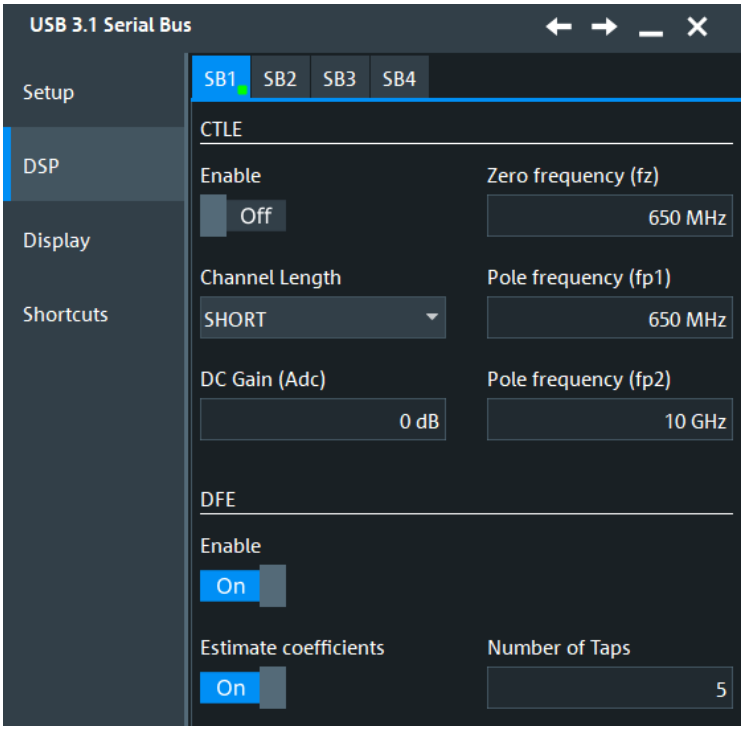
Show thresholds
If enabled, the threshold levels are displayed in the diagram.

Remote command:
`BUS<m>:THReshold` on page 1671

13.21.2.2 USB 3.1 DSP settings

Access: [Bus] > "Setup" tab > "Protocol = USB 3.1" > "DSP" tab

At this tab, you can enable and define the equalization techniques used for the equalization at the reciever side of the signal.



CTLE
Continuous time linear equalization (CTLE) is an equalization that can be used for improving the signal performance. It boosts the high frequency noise and the signal. Thus it helps reduce the inter-symbol interference (ISI) on the reciever side.

Enable ← CTLE
Enables the CTLE equalization.

Remote command:
`BUS<m>:CTLE:ENBLe` on page 2190

Zero frequency (fz) ← CTLE
Sets the zero frequency for the transfer function of the CTLE algorithm.

Remote command:
`BUS<m>:CTLE:Z1FReq` on page 2191

Pole frequency (wp1)/(wp2) ← CTLE
Sets the pole 1/2 frequency for the transfer function of the CTLE algorithm.

Remote command:
`BUS<m>:CTLE:P1FReq` on page 2190
`BUS<m>:CTLE:P2FReq` on page 2191

DC Gain (A_{DC}) ← CTLE
Sets the adjustable DC gain. This value can be chosen according to the channel loss characteristics. For low-loss channels, you can set a value close to 0 dB.

Remote command:
`BUS<m>:CTLE:DCGain` on page 2190

Channel Length ← CTLE
Selects the USB Super Speed channel length.

Remote command:
`BUS<m>:USBThree:PROTocol` on page 2192

DFE
Defines the settings of the decision feedback equalization (DFE). DFE correction is needed when the channel is distosrted.

Enable ← DFE
Enables the use of the decision feedback equalization.

Remote command:
`BUS<m>:USBThree:DFE:ENBLe` on page 2192

Estimate coefficients ← DFE
Enables the use of estimate coefficients.

Remote command:
`BUS<m>:USBThree:DFE:ESTCoeff` on page 2192

Coefficient ← DFE
Sets the value of the coefficient.

Remote command:
`BUS<m>:USBThree:DFE:COEFficient` on page 2191

Number of Taps ← DFE
Sets a number of taps for the decision feedback filter.

Remote command:
`BUS<m>:USBThree:DFE:TCONt` on page 2192

13.21.2.3 Configuring USB 3.1 signals

For configuration, assign the lines to the input channels and define the active states and the logical thresholds.

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. In the "Setup" tab, set "Protocol"="USB3.1".
4. Select the "Protocol type".
5. Switch back to the "Setup" tab.
6. Set the logical thresholds: Either enter user-defined values directly in the "Threshold" fields, or set the thresholds to the middle reference levels with "Auto thresholds".
7. Enable "Decode".

For details on configuration settings, see Chapter 13.21.2.1, "USB 3.1 configuration settings", on page 849.

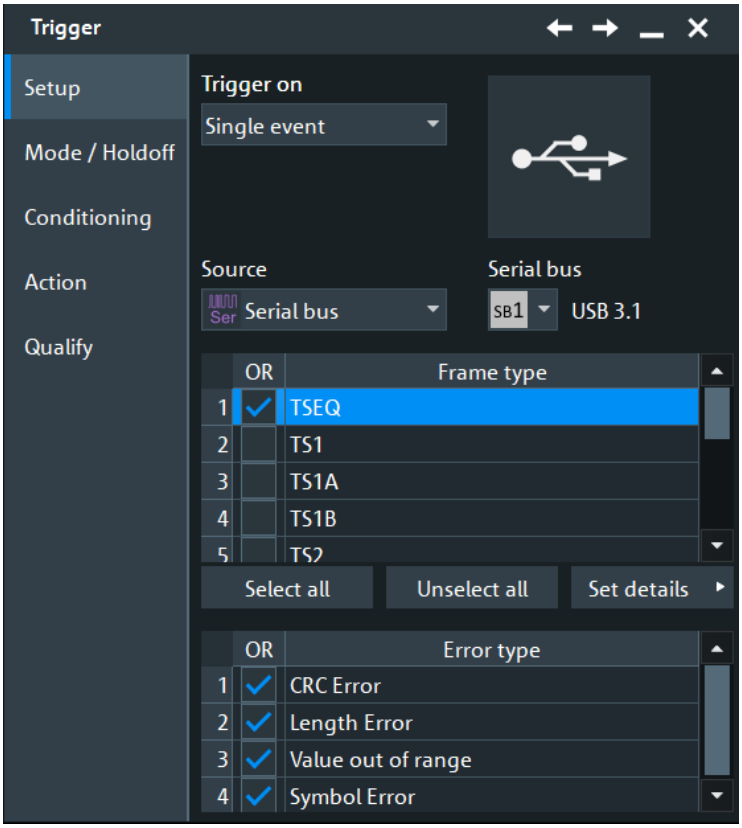
13.21.3 USB 3.1 trigger

If you need information on how to get started with triggering on USB 3.1 signals, see Chapter 13.21.3.2, "Triggering on USB 3.1", on page 856. Otherwise proceed with the USB 3.1 trigger settings.

13.21.3.1 USB 3.1 trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"

In this section, all trigger settings for USB 3.1 are described. The user interface of the instrument guides you through the trigger setup.



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Protocol type
Selects the protocol type for the USB 3.1 analysis.

Frame type
Selects the frame type for USB 3.1 analysis. For the available frame types, see "Frame types" on page 842.

You can define individual checking parameters for the fields listed in the "Set details" dialog.

The trigger instant is the last criterion that is fulfilled.

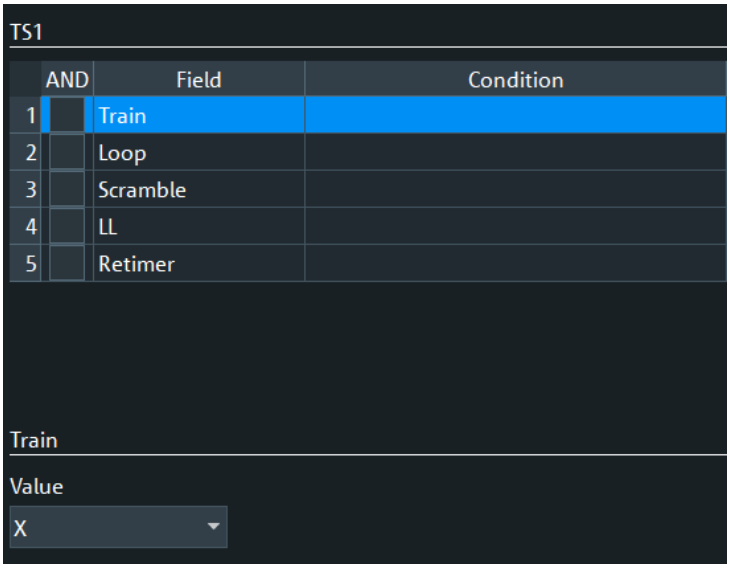
Remote command:
`TRIGger<m>:USBThree:FRENable` on page 2200
`TRIGger<m>:USBThree:FRAMe<n>:ENABle` on page 2200

Error types: Item name, Enable
The table lists the error types you can trigger on: "CRC Error", "Length Error", "Value out of range", "Symbol Error". Enable any error that you want to trigger on in the "OR" column.

Remote command:
`TRIGger<m>:USBThree:ERENable` on page 2199
`TRIGger<m>:USBThree:ERRor<n>:ENABle` on page 2199

Set details
Select the frame type first, and then open the "Set details" dialog.

In this dialog to define the individual parameters for the fields of the selected frame.



Field name / Condition Summary / Enable ← Set details

This table lists the field numbers and names in the selected frame together with a summary of the user settings of checking conditions for each field and a checkbox to enable the checking.

It also shows the user settings of checking conditions for each field and a checkbox to enable the checking.

Select a field in the table to specify the checking conditions for this field in the "Set details" dialog.

For an overview of frames and fields, see Frame types.

Remote command:

`TRIGger<m>:USBThree:FIENable` on page 2200

`TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:ENABle` on page 2200

Data

Defines for the selected field, how a data check is executed.

"Condition"	Sets the comparison condition to a specific value or a range.
"Min"	Specifies the value or sets the start value of a range.
"Max"	Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:

`TRIGger<m>:USBThree:DOPerator` on page 2201

`TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DOPerator` on page 2201

`TRIGger<m>:USBThree:DMIN` on page 2201

`TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DMIN` on page 2201

`TRIGger<m>:USBThree:DMAX` on page 2202

`TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DMAX` on page 2202

Data BitState
Defines the data bit state to be checked for the selected field. Permitted bit states are "1", "0" or "X" (do not care).

Remote command:
`TRIGger<m>:USBThree:BIT` on page 2202
`TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:BIT` on page 2202

Index
The trigger index setup consists of a comparison condition and one or two address patterns.

"Condition"	Sets the comparison condition to a specific value or a range.
"Min"	Specifies the value or sets the start value of a range.
"Max"	Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:USBThree:IOPerator` on page 2202
`TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IOPerator` on page 2202
`TRIGger<m>:USBThree:IMIN` on page 2203
`TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IMIN` on page 2203
`TRIGger<m>:USBThree:IMAX` on page 2203
`TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IMAX` on page 2203

13.21.3.2 Triggering on USB 3.1

Prerequisite: A bus is configured for the USB 3.1 signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Select the "Protocol Type".
5. If you trigger on errors, enable the error types you want to find.
6. Select the frame types you want to trigger on.
7. For some frame types, you can define the frame fields.
   For information on the configuration settings, see Chapter 13.21.3.1, "USB 3.1 trigger settings", on page 853.

13.21.4 USB 3.1 decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".

2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

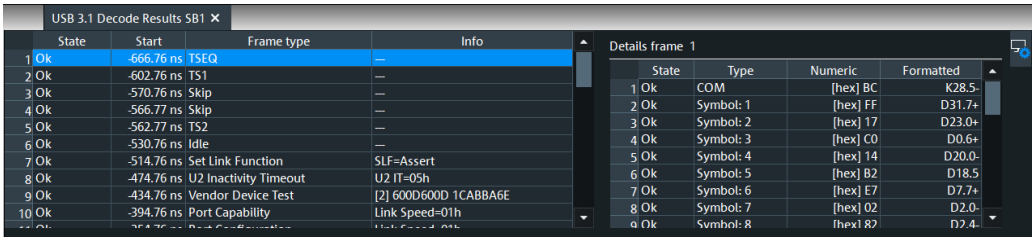


Table 13-40: Content of the Decode results table

Column	Description
Frame	Frame number
State	Overall state of the frame, for example "OK" or "Incomplete"
Start	Start time of the frame
Frame Type	Type of the frame
Info	Specific frame information, for example result of the Link Speed Measurement = 5.0 Gb/s, or content of the Data Packet Payload

Enable "Details" in the decode table to display a more detailed analysis of the selected frame.

Table 13-41: Content of the Details frame table

Column	Description
State	Overall state of the field
Start	Start time of the field
Stop	Stop time of the field
Type	Type (name) of field

Column	Description
Numeric value	The numeric value of the field
Formatted	Formatted content of the field, for example HPSTART "F7" = end packet framing (EPF)

Remote commands

Remote control commands for retrieving decode results are described in Chapter 24.17.22, "USB 3.1 (option R&S RTO6-K570)", on page 2188.

Remote command:

- `BUS<m>:USBThree:RESult:FCOunt?` on page 2204
- `BUS<m>:USBThree:RESult:FRAMe<n>:INFO?` on page 2204
- `BUS<m>:USBThree:RESult:FRAMe<n>:TYPE?` on page 2204
- `BUS<m>:USBThree:RESult:FRAMe<n>:STATe?` on page 2205
- `BUS<m>:USBThree:RESult:FRAMe<n>:STARt?` on page 2205
- `BUS<m>:USBThree:RESult:FRAMe<n>:STOP?` on page 2206
- `BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:TYPE?` on page 2206
- `BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:STATus?` on page 2206
- `BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:STARt?` on page 2207
- `BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:STOP?` on page 2207
- `BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:VAL?` on page 2207
- `BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:FVAL?` on page 2208

13.21.5 Search on decoded USB 3.1 data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 11, "Search functions", on page 419.

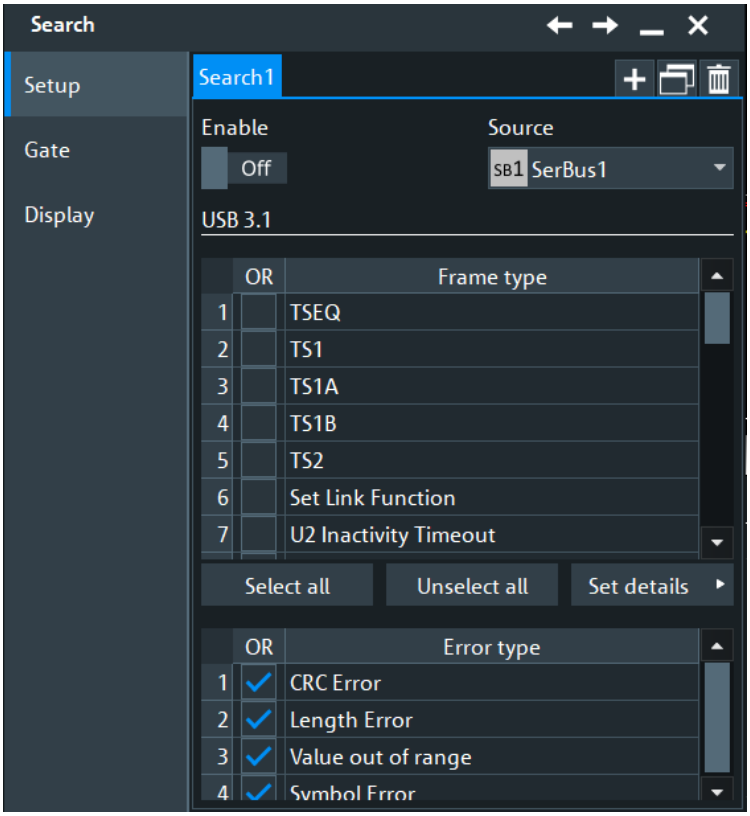
If you need information on how to get started with searching USB 3.1 data, see Chapter 13.21.5.3, "Searching USB 3.1 data", on page 860. Otherwise proceed with the USB 3.1 search setup.

13.21.5.1 USB 3.1 search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"

Search criteria

Use the "Search criteria" dialog to define the search type. If you search for frame content, define the frame type in which data is to be searched.



You can define individual search parameters for the fields of the selected frame in the table below the "Search criteria". To specify these parameters, select a field in the table and define the data and/or index operators and values, or the bit state.

For a description of how to set the search conditions, see Chapter 13.21.3.1, "USB 3.1 trigger settings", on page 853.

Remote command:

`SEARch:TRIGger:USBThree:ERENable` on page 2209
`SEARch:TRIGger:USBThree:ERRor<m>:ENABle` on page 2209
`SEARch:TRIGger:USBThree:FRENable` on page 2209
`SEARch:TRIGger:USBThree:FRAMe<m>:ENABle` on page 2209
`SEARch:TRIGger:USBThree:FIENable` on page 2210
`SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:ENABle` on page 2210
`SEARch:TRIGger:USBThree:DOPerator` on page 2210
`SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DOPerator` on page 2210
`SEARch:TRIGger:USBThree:DMIN` on page 2211
`SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DMIN` on page 2211
`SEARch:TRIGger:USBThree:DMAX` on page 2211
`SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DMAX` on page 2211
`SEARch:TRIGger:USBThree:BIT` on page 2212
`SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:BIT` on page 2212

SEARch:TRIGger:USBThree:IOPerator on page 2212
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IOPerator on page 2212
SEARch:TRIGger:USBThree:IMIN on page 2213
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMIN on page 2213
SEARch:TRIGger:USBThree:IMAX on page 2213
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMAX on page 2213

13.21.5.2 USB 3.1 search results

To get search results, "Enable" the search in the "Control" section of the "Search Setup" dialog. You can minimize, shift or close the search dialog to better see the "Search Results" table.

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands:

- SEARch:RESult:USBThree:FCOunt? on page 2215
- SEARch:RESult:USBThree:FRAMe<m>:TYPE? on page 2214
- SEARch:RESult:USBThree:FRAMe<m>:INFO? on page 2214
- SEARch:RESult:USBThree:FRAMe<m>:STATe? on page 2215
- SEARch:RESult:USBThree:FRAMe<m>:STARt? on page 2215
- SEARch:RESult:USBThree:FRAMe<m>:STOP? on page 2216
- SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:TYPE? on page 2216
- SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:STATus? on page 2216
- SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:STARt? on page 2217
- SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:STOP? on page 2217
- SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:VAL? on page 2218
- SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:FVAL? on page 2218

13.21.5.3 Searching USB 3.1 data

Prerequisite: A serial bus is configured for the USB 3.1 signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Tap "Setup search".

3. If the dialog box does not contain a search entry, tap the "Add" icon to create one, as described in "To create a user-defined search" on page 434.
4. Tap "Source" and select the serial bus that is set to USB 3.1 (e.g. "SB1", unless already selected).

   The search dialog for USB3 protocol analysis is opened.
5. Specify search criteria according to Chapter 13.21.5.1, "USB 3.1 search setup", on page 858.
6. To acquire a waveform, press [Single].

   The R&S RTO6 performs a USB 3.1 decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).
7. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

   The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.22 USBPD (option R&S RTO6-K570)

R&S RTO6-K570 is a firmware option that enables the R&S RTO6 to analyze Universal Serial Bus Power Delivery (USBPD) signals.

For analysis, USBPD encoded signals can be triggered, decoded and searched.

This chapter describes:

- The USB power delivery protocol....861
- USBPD configuration....863
- USBPD trigger....866
- USBPD decode results....870
- Search on decoded USBPD data....872

13.22.1 The USB power delivery protocol

The requirements on the USB have changed in the last years with the need of providing power through the USB port additionally to the data transfer. The USBPD specification aims to define standard for optimizing the power usage through the USB for the needs of the users.

USBPD characteristics

Main characteristics of USBPD are:

- Power direction is not fixed
- Negotiation of required power amount between devices

- Alternate modes can be defined through vendor defined messages, which allows for USB connector pins to be used for purposes other than USB

Message types

In the USBPD protocol, a power delivery connection can be made between a port that supplies power (source) and a port that consumes power (sink). They communicate with each other through messages. The USBPD specification defines three message types:

- Control messages: 16-bit messages used to control the messages between the port partners or transfer messages with no extra data. A control message consists of a message header and a CRC.
- Data messages: 48 bit to 240 bit messages used to transfer information between port partners. A data message consists of a message header and several data objects. The information that a data object carries is defined by the message type of the message header, see Table 13-42.
- Extended messages: can have a different length up to the defined maximum length of an extended message. It is used to transfer information between port partners. The information that the extended message carries is defined by the message type of the message header, see Table 13-42.

Frame packet types

All frame types are listed below. The frames listed above consist of individual sets of fields. Some frames only contain one field, others are much longer.

The frame types are sorted according to the message type.

Table 13-42: Frame packet types

SCPI	Description	Message type
ALRT	Alert message	Data
BATT	Battery status	Data
BIST	Built in self-test	Data
RQST	Request	Data
SINK	Sink capabilities message	Data
SRC	Source capabilities message	Data
VEND	Vendor defined message	Data
CTRL	Control	Control
DATA	Data	Data
LOWP	Low power	Low power
TEST	Test frame	Test
RESet	Reset frame	Reset
XBAC	Battery capabilities	Extended
XFRS	Firmware update response	Extended

SCPI	Description	Message type
XFRQ	Firmware update request	Extended
XGBC	Get battery cap	Extended
XGMI	Get manufacturer info	Extended
XGBS	Get battery status	Extended
XMFI	Manufacturer info	Extended
XMSG	Message	Extended
XSRC	Sources capabilities message	Extended
XSRS	Security response	Extended
XSRQ	Security request	Extended
XSTA	Status	Extended

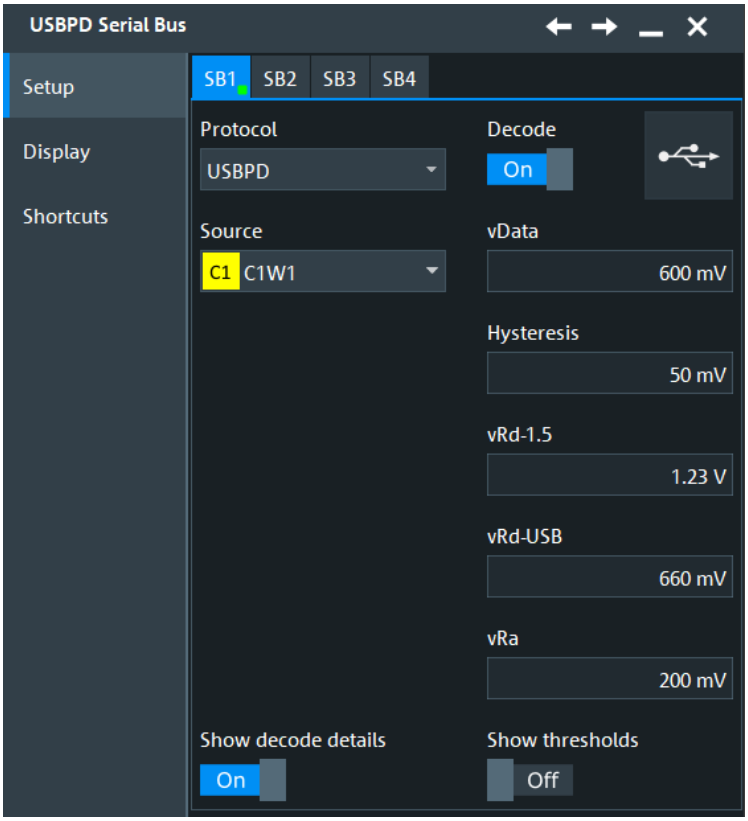
More information on the USBPD protocol, including all specifications down to the field contents, is available in the "Universal Serial Bus Power Delivery Specification" documentation. Refer to the online resources at www.usb.org.

13.22.2 USBPD configuration

13.22.2.1 USBPD configuration settings

Access: [Bus] > "USBPD" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Data
Defines the source settings for the data signal. You can select analog channels "C1" to "C4", and also active math and reference waveforms as source. If you want to trigger on the serial bus, analog channel sources are required.

Remote command:
`BUS<m>:USBPd:SOURce` on page 2219

vData
Sets the threshold value of the data.

Remote command:
`BUS<m>:USBPd:THReshold` on page 2219

Hysteresis
Sets a value for the hysteresis of the data.

Remote command:
`BUS<m>:USBPd:HYSTeresis` on page 2219

Current advertisment thresholds
The signal level provides information about the current advertisement between the bursts. These thresholds determine the levels at which the current advertisement modes are defined.

"vRd-1.5" Sets the threshold at USB Type-C current of 1.5 A.

"vRd-USB" Sets the threshold at default USB Type-C current.

"vRa" Sets the threshold for the low current.

Remote command:
`BUS<m>:USBPd:THRBottom` on page 2220
`BUS<m>:USBPd:THRMid` on page 2220
`BUS<m>:USBPd:THRTop` on page 2220

Show decode detail
If enabled, the data words are broken down into subframes. If not enabled the data words are displayed as 32-bit data words.

If the "Show decode detail" is enabled, you cannot do a search and trigger on USBPD frames.

Remote command:
`BUS<m>:USBPd:DETail` on page 2219

Show thresholds
If enabled, the threshold levels are displayed in the diagram.

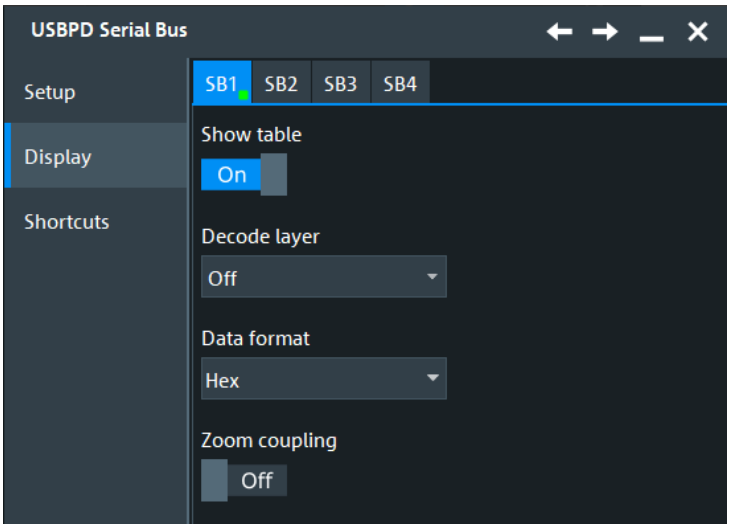
Remote command:
`BUS<m>:THReshold` on page 1671

13.22.2.2 Display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

To enhance the decode possibilities of the USBPD protocol, you can use an additional setting in the "Display" tab: "Decode layer".



Decode layer
Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

Available are:
- "Edges"
- "Bits"
- "4b5b Symbols"

13.22.2.3 Configuring the USBPD signals

For configuration, assign the lines to the input channels and define the logical thresholds and the hysteresis.

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. In the "Setup" tab, set "Protocol"="USBPD".
4. Select the data source.
5. Enter the "Thresholds".
6. Enable "Decode".

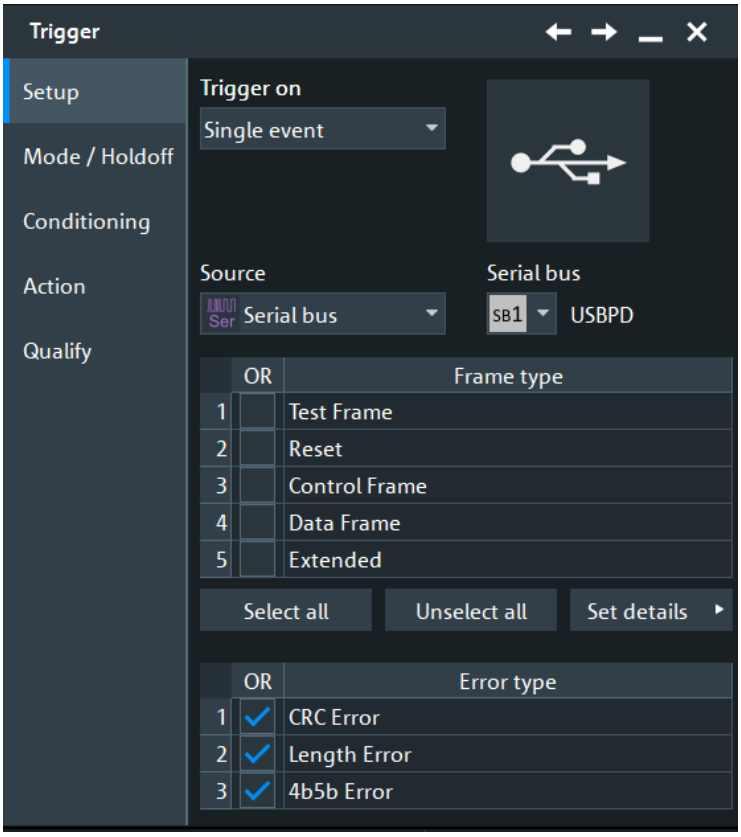
13.22.3 USBPD trigger

If you need information on how to get started with triggering on USBPD signals, see Chapter 13.22.3.2, "Triggering on USBPD", on page 869. Otherwise proceed with the USBPD trigger settings.

13.22.3.1 USBPD trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"

In this section, all trigger settings for USBPD are described. The user interface of the instrument guides you through the trigger setup.



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Frame Type

Selects the frame type for the USBPD analysis. For the available packet types, see "Frame packet types" on page 862.

You can define individual checking parameters for the fields listed of the frame.

To specify these parameters, select a field from this list and open "Set details". You can define the data and/or index operators and values, or the bit state.

The trigger instant is the last criterion that is fulfilled.

Remote command:

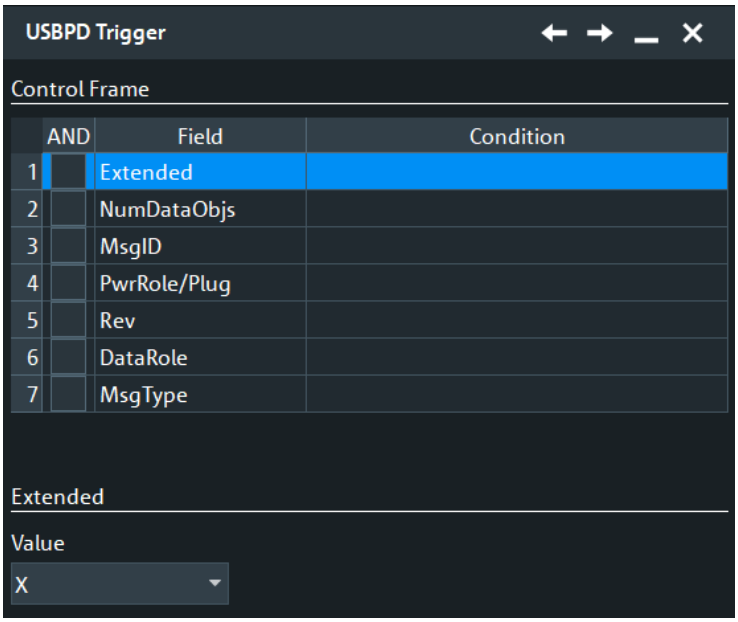
`TRIGger<m>:USBPd:FRENable` on page 2223

`TRIGger<m>:USBPd:FRAMe<n>:ENABle` on page 2223

Set details

Select the frame type first, and then open the "Set details" dialog.

In this dialog to define the individual parameters for the fields of the selected frame.



Enable/ Field name/ Condition ← Set details

This table lists the field numbers and names in the selected frame together with a summary of the user settings of checking conditions for each field and a checkbox to enable the checking.

Select a field in the table to specify the checking conditions for this field in the "Data", "BitState" and/or "Index" dialog (whichever applies). The condition is only applied, and the "Condition Summary" is only shown in the table, if "Enable" is checked.

For an overview of frames and fields, see "Frame packet types" on page 862.

Remote command:

`TRIGger<m>:USBPd:FIENable` on page 2223

`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:ENABle` on page 2223

Error > Enable /Item

The table lists the error types you can trigger on: "CRC Error", "Length Error", "4b5b Error". Enable any error that you want to trigger on in the "OR" column.

Remote command:

`TRIGger<m>:USBPd:ERENable` on page 2222

`TRIGger<m>:USBPd:ERRor<n>:ENABle` on page 2222

Data

Sets the data to be triggered on. The trigger data setup consists of a comparison condition and one or two data patterns.

"Condition"	Sets the comparison condition to a specific value or a range.
"Min"	Specifies the value or sets the start value of a range.

"Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:USBPd:DOPerator` on page 2225
`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DOPerator` on page 2225
`TRIGger<m>:USBPd:DMAX` on page 2224
`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMAX` on page 2224
`TRIGger<m>:USBPd:DMIN` on page 2225
`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMIN` on page 2225

Data BitState
Defines the bit state to be checked for the selected field. Permitted bit states are "1", "0" or "X" (do not care).

Remote command:
`TRIGger<m>:USBPd:BIT` on page 2224
`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:BIT` on page 2224

Index
Defines for the selected field, how an index check is executed.

"Condition" Defining a specific index or an index range requires to set the operator to one of the following conditions: equal, in range.

"Min" Specifies the index or sets the start value of an index range.

"Max" Sets the the end value of an index range, if "Condition" is set to `INRange`.

Remote command:
`TRIGger<m>:USBPd:IOPerator` on page 2226
`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IOPerator` on page 2226
`TRIGger<m>:USBPd:IMAX` on page 2225
`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMAX` on page 2225
`TRIGger<m>:USBPd:IMIN` on page 2226
`TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMIN` on page 2226

13.22.3.2 Triggering on USBPD

Prerequisite: A bus is configured for the USBPD signal to be analyzed.

In order to be able to trigger on an USBPD data, the "Show decode detail" field in the "Configuration" tab of the protocol setup should be disabled.

For the basic trigger settings, proceed in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Tap "Serial bus" and select the serial bus that is set to USBPD.

5. If you trigger on errors, enable the error types you want to find.
6. Select the frame types you want to trigger on.
7. For some frame types, you can define the frame fields.
   For information on how to proceed with the configuration settings, see Chapter 13.22.3.1, "USBPD trigger settings", on page 866.

13.22.4 USBPD decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Serial Bus" dialog > "Setup" tab, enable "Decode".
2. In the "Serial Bus" dialog > "Display" tab, select additional result display settings: "Show table" and "Show binary".
3. If required, enable "Zoom coupling"

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The example in Figure 13-48 shows a simulated USBPD message. Among the long line of decoded frames, the zoom has selected a "Control" frame.

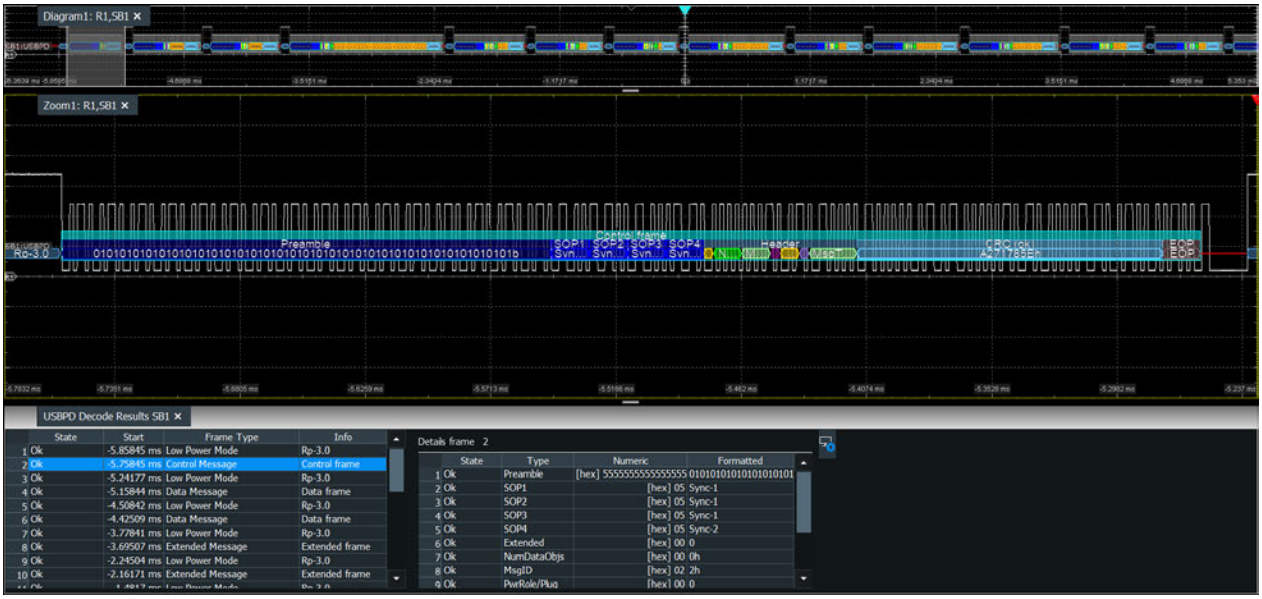


Figure 13-48: USBPD protocol: decoded signal

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

Table 13-43: Content of the Decode results table

Column	Description
State	Overall state of the frame
Start	Time of frame start
Frame Type	Type of the frame
Info	Information about the frame

Table 13-44: Content of the Details frame table

Column	Description
State	Overall state of the field
Start	Start time of the field
Type	Type (name) of field
Numeric value	The numeric value of the field
Formatted	Formatted content of the field

Remote commands

Remote commands are described in Chapter 24.17.23.3, "Decode results", on page 2227.

13.22.5 Search on decoded USBPD data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

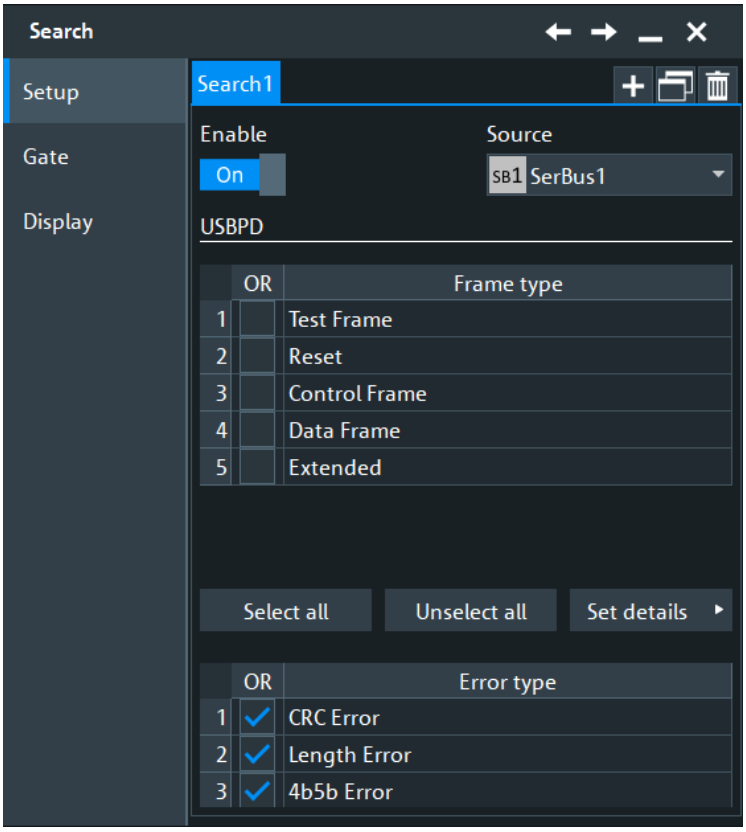
For general information on how to handle the search functionality, see Chapter 11, "Search functions", on page 419.

If you need information on how to get started with searching USBPD data, see Chapter 13.22.5.3, "Searching USBPD data", on page 874. Otherwise proceed with the USBPD search setup.

13.22.5.1 USBPD search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"

Search
In the "Search" dialog to define the frame type in which data is to be searched.



You can define individual search parameters for individual fields in the "Set details" dialog. To specify these parameters, select a field in this table and define the data and/or index operators and values, or the bit state.

For a description of how to set the search conditions, see Chapter 13.22.3.1, "USBPD trigger settings", on page 866.

Remote command:
`SEARch:TRIGger:USBPd:ERENable` on page 2232
`SEARch:TRIGger:USBPd:ERRor<m>:ENABle` on page 2232
`SEARch:TRIGger:USBPd:BIT` on page 2234
`SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:BIT` on page 2234
`SEARch:TRIGger:USBPd:DMAX` on page 2234
`SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMAX` on page 2234
`SEARch:TRIGger:USBPd:DMIN` on page 2235
`SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMIN` on page 2235
`SEARch:TRIGger:USBPd:DOPerator` on page 2235
`SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DOPerator` on page 2235
`SEARch:TRIGger:USBPd:IMAX` on page 2236
`SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMAX` on page 2236
`SEARch:TRIGger:USBPd:IMIN` on page 2236
`SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMIN` on page 2236
`SEARch:TRIGger:USBPd:IOPerator` on page 2237
`SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IOPerator` on page 2237

13.22.5.2 USBPD search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog to better see the "Search Results" table.

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands:

- `SEARch:RESult:USBPd:FCOunt?` on page 2238
- `SEARch:RESult:USBPd:FRAMe<m>:DATA?` on page 2238
- `SEARch:RESult:USBPd:FRAMe<m>:TYPE?` on page 2242
- `SEARch:RESult:USBPd:FRAMe<m>:INFO?` on page 2241
- `SEARch:RESult:USBPd:FRAMe<m>:STATe?` on page 2241
- `SEARch:RESult:USBPd:FRAMe<m>:STARt?` on page 2241
- `SEARch:RESult:USBPd:FRAMe<m>:STOP?` on page 2242
- `SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:TYPE?` on page 2240

- `SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STATus?` on page 2239
- `SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STARt?` on page 2239
- `SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STOP?` on page 2239
- `SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:VAL?` on page 2240
- `SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:FVAL?` on page 2238

13.22.5.3 Searching USBPD data

Prerequisite: A serial bus is configured for the USBPD signal to be decoded and analyzed.

To be able to search for a USBPD data, the "Show decode detail" field in the "Configuration" tab of the protocol setup should be disabled. If "Show decode detail" is enabled, the "Enable" search button is disabled.

The search for events is set up in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Tap "Setup search".
3. If the dialog box does not contain a search entry, tap the "Add" icon to create one, as described in "To create a user-defined search" on page 434.
4. Tap "Source" and select the serial bus that is set to USBPD (e.g. "SerBus1", unless already selected).

   The search dialog for USBPD protocol analysis is opened.
5. Specify search criteria according to Chapter 13.22.5.1, "USBPD search setup", on page 872.
6. To acquire a waveform, press [Single].

   The R&S RTO6 performs a USBPD decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).
7. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

   The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.23 D-PHY (option R&S RTO6-K580)

The D-PHY is a specification developed by the Mobile Industry Processor Interface (MIPI) alliance as a standard for the communication of high-speed components, like cameras and displays, within mobile devices.

- D-PHY basics....875
- D-PHY configuration....875
- D-PHY trigger....879
- D-PHY decode results....885
- Search on decoded D-PHY data....886

13.23.1 D-PHY basics

A D-PHY interface consists of one clock lane and up to four data lanes. The D-PHY data lanes have two operational modes:

- A high speed mode (HS): differential signal with a data rate of 80 Mbps to 1.5GHz.This mode is used for the transmission of large volumes of information.
- A low power mode (LP): single-ended signal with a data rate < 10 Mbps. This mode is used for conserving power .

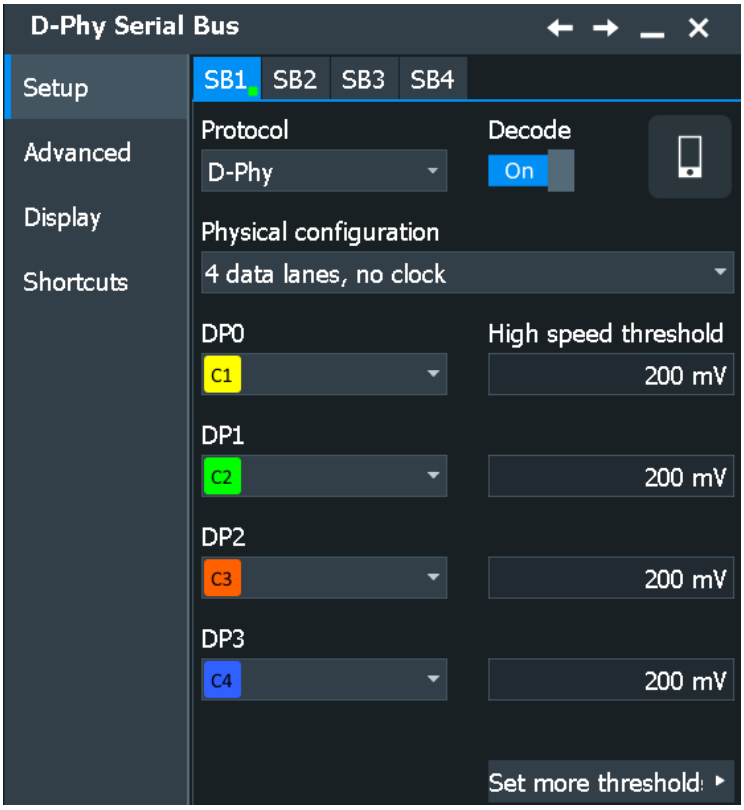
D-PHY provides a framework for other protocols such as the Display Serial Interface (DSI) and Camera Serial Interface (CSI-2).

13.23.2 D-PHY configuration

13.23.2.1 D-PHY configuration settings

Access: [Bus] > "D-PHY" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Variant
Selects the protocol running on the interface. You can select between the Display Serial Interface (DSI) and Camera Serial Interface (CSI-2).

Remote command:
`BUS<m>:DPHY:VARiant` on page 2245

Physical configuration
Selects the number of data lanes and if the clock lane and low power are available.

Remote command:
`BUS<m>:DPHY:CONFig` on page 2243

Source Connection
Defines the source and the type of probe used for the respective lane.

Permitted source selections are none, the analog, mathematical, and reference channels.

CP ← Source Connection
Sets the source of the clock lane.

Remote command:
`BUS<m>:DPHY:CP:SOURce` on page 2244

DP0/DN0 ← Source Connection
Selects the source of the low-power data lanes.

The "DP0" lane has to be present all the time, hence "Source DP0">"None" is not allowed.

Remote command:
`BUS<m>:DPHY:DPZero:SOURce` on page 2246
`BUS<m>:DPHY:DNZero:SOURce` on page 2246

DP1/DP2/DP3 ← Source Connection
Selects the source of the high-speed data lanes.

Remote command:
`BUS<m>:DPHY:DPONe:SOURce` on page 2246
`BUS<m>:DPHY:DPTWo:SOURce` on page 2246
`BUS<m>:DPHY:DPTHree:SOURce` on page 2246

Probe ← Source Connection
Selects the type of probe used for the respective lane. The D-PHY low-power mode uses a single-ended signal, while the high-speed mode signaling is differential.

Remote command:
`BUS<m>:DPHY:CP:PROBe` on page 2244
`BUS<m>:DPHY:DNZero:PROBe` on page 2247
`BUS<m>:DPHY:DPZero:PROBe` on page 2247
`BUS<m>:DPHY:DPONe:PROBe` on page 2247
`BUS<m>:DPHY:DPTWo:PROBe` on page 2247
`BUS<m>:DPHY:DPTHree:PROBe` on page 2247

Data Rate
Sets a data rate.

Remote command:
`BUS<m>:DPHY:DRATe` on page 2244
`BUS<m>:DPHY:DSPData` on page 2245

HS Threshold
Sets the threshold value for the digitization of high-speed data line.

Remote command:
`BUS<m>:DPHY:DPZero:HSPeed:THReshold` on page 2247
`BUS<m>:DPHY:DPONe:HSPeed:THReshold` on page 2247
`BUS<m>:DPHY:DPTWo:HSPeed:THReshold` on page 2247
`BUS<m>:DPHY:DPTHree:HSPeed:THReshold` on page 2247

Set thresholds
Opens a dialog to set the threshold values.

HS Hysteresis ← Set thresholds
Sets a value for the hysteresis of the respective lane.

Remote command:
`BUS<m>:DPHY:CP:HSPeed:HYSTeresis` on page 2247
`BUS<m>:DPHY:DPZero:HSPeed:HYSTeresis` on page 2247
`BUS<m>:DPHY:DPONe:HSPeed:HYSTeresis` on page 2247
`BUS<m>:DPHY:DPTWo:HSPeed:HYSTeresis` on page 2247
`BUS<m>:DPHY:DPTHree:HSPeed:HYSTeresis` on page 2247

Low power thresholds ← Set thresholds
Sets the thresholds for the low-power mode.

DP0 High / DN0 High ← Low power thresholds ← Set thresholds
Sets the high-power threshold value for the respective lane.

Remote command:
`BUS<m>:DPHY:DNZero:LPOWer:THUPper` on page 2246
`BUS<m>:DPHY:DPZero:LPOWer:THUPper` on page 2246

DP0 Low / DN0 Low ← Low power thresholds ← Set thresholds
Sets the low-power threshold value for the respective lane.

Remote command:
`BUS<m>:DPHY:DNZero:LPOWer:THLower` on page 2246
`BUS<m>:DPHY:DPZero:LPOWer:THLower` on page 2246

Couple thresholds ← Set thresholds
Enables coupling, i.e. the same threshold and hysteresis value is used for all lanes.

Remote command:
`BUS<m>:DPHY:THCoupling` on page 2245

13.23.2.2 Display settings

Access: [Bus] > "Configuration" tab > "Protocol = D-PHY" > "Display" tab

To enhance the decode possibilities of the D-PHY protocol, you can use an additional setting in the "Display" tab: "Decode layer".

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

Decode layer
Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

"HS Edges"	The high-speed edges of each lane, showing the transitions in high-speed mode.
"HS Binary"	All high-speed bits of each lane, whether they are part of a burst or not.

"HS Burst Bits" The filtered high-speed bits of each lane , including only bits that are part of a burst.

"HS Burst Words" 8 bits per word in a high-speed burst of each lane.

"HS Merged Bytes" The words from the previous layer after they have been merged into one honeycomb.

"HS Merged Words" The bytes that are combined/split into words.

"LP Edges" The combined edge display of DP0 and DN0.

"LP States" The low-power states LP00, LP01, LP10, LP11 of DP0 and DN0.

Show multiple lanes
Enables the display of the results in multiple lanes.

Remote command:
`BUS<m>:DPHY:SMLanes` on page 2248

13.23.2.3 Configuring the D-PHY signals

For configuration, assign the lines to the input channels and define the logical thresholds and the hysteresis.

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. In the "Setup" tab, select the protocol: "D-PHY".
4. Tap the "Variant" button, and select the protocol.
5. Enter the number of "Data Lanes".
6. Select the source and type of "Probe", for each lane.
7. Enter the "HS threshold" and the "Hysteresis" for each data lane.
8. Enter the low-power thresholds.
9. Enable "Decode".

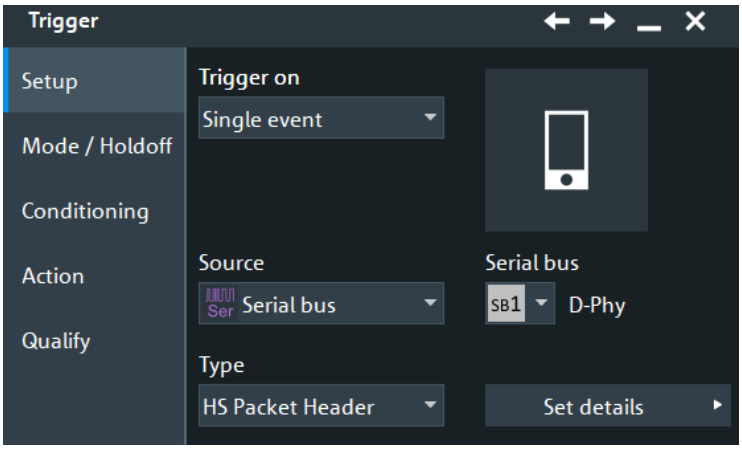
13.23.3 D-PHY trigger

13.23.3.1 D-PHY trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"

In this section, all trigger settings are described. Their availability on the instrument depends on the selected USB protocol type and trigger type. The user interface of the instrument displays only appropriate settings and guides you through the trigger setup.

For a list of supported trigger conditions, refer to data sheet.



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Serial bus

Selects the serial bus to be triggered on. Make sure to select the correct bus before you enter the settings.

To trigger on a serial bus, the signals sources must be channel signals. If the data or clock source is a math or reference waveform, you cannot trigger on that bus.

Remote command:

`TRIGger<m>:SOURce:SBSelect` on page 1675

Protocol

Defines the protocol type of the selected serial bus.

Remote command:

`BUS<m>:TYPE` on page 1670

Type

Selects the trigger type for the D-PHY analysis.

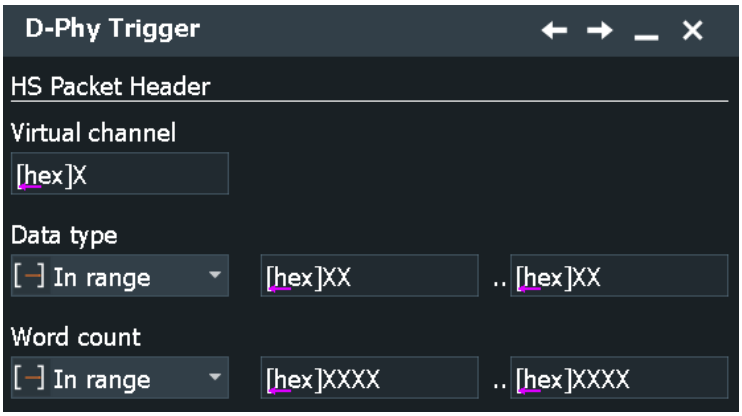
Remote command:

`TRIGger<m>:DPHY:TYPE` on page 2249

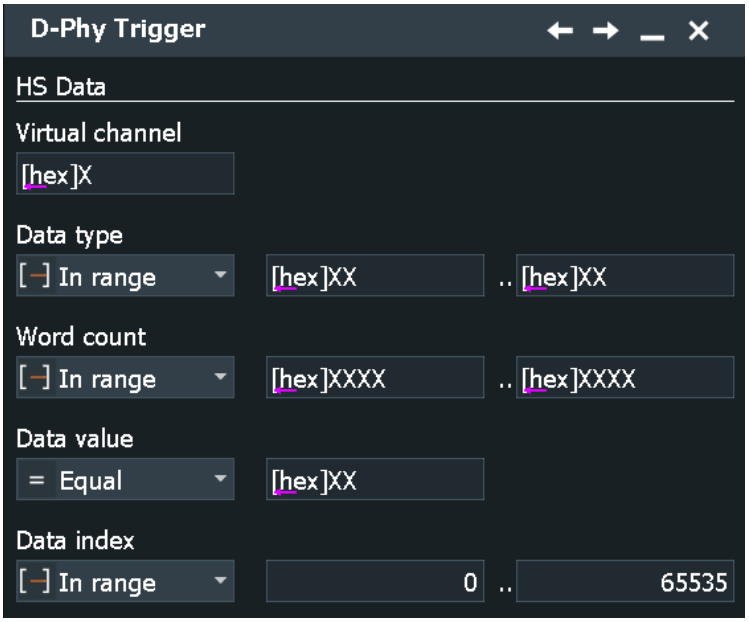
HS Start of Packet ← Type
Sets the trigger to the transmission start of a high-speed packet.

HS End of Packet ← Type
Sets the trigger to the transmission end of a high-speed packet.

HS Packet Header ← Type
Sets the trigger to a packet header of a high-speed package. The header consists of a data identifier (containing the Virtual Channel and the Data Type) and a Word Count. You can specify the values each part of the packet header.



HS Data ← Type
Sets the trigger to a specified high-speed data.



LP Escape Mode ← Type
Sets the trigger to an escape mode event.

LP Lane Turnaround ← Type
Sets the trigger to a low-power turnaround, a reversion of the transmission direction.

LP HS Request ← Type
Sets the trigger to a high-speed request.

Virtual Channel
Triggers on a specific virtual channel, an independent data stream for up to four peripherals.

Remote command:
`TRIGger<m>:DPHY:HSVC` on page 2254

Data Type
Sets the specified data type to be triggered on. The data type setup consists of the condition and one or two data patterns.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Condition"	Sets the comparison condition to a specific value or a range.
"Data Min/ Data"	Specifies the value or sets the start value of a range.
"Data Max"	Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:DPHY:DTYPe:CONDition` on page 2251
`TRIGger<m>:DPHY:DTYPe:MAX` on page 2251
`TRIGger<m>:DPHY:DTYPe:MIN` on page 2251

Word Count
Sets the specified specific word count to be triggered on. The word count setup consists of the condition and one or two data patterns.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Condition"	Sets the comparison condition to a specific value or a range.
"Data Min/ Data"	Specifies the value or sets the start value of a range.
"Data Max"	Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:DPHY:WCOunt:CONDition` on page 2254
`TRIGger<m>:DPHY:WCOunt:MAX` on page 2254
`TRIGger<m>:DPHY:WCOunt:MIN` on page 2255

Data Value
Set the specified data value to be triggered on. The data value setup consists of the condition and one or two data value patterns.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Condition"	Sets the comparison condition to a specific value or a range.

"Data Min/Data" Specifies the value or sets the start value of a range.

"Data Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
For "Type">"HS Data":
`TRIGger<m>:DPHY:DATA:CONDition` on page 2249
`TRIGger<m>:DPHY:DATA:MAX` on page 2250
`TRIGger<m>:DPHY:DATA:MIN` on page 2250
For "Type">"LP Escape Mode":
`TRIGger<m>:DPHY:ESDTa:CONDition` on page 2252
`TRIGger<m>:DPHY:ESDTa:MAX` on page 2253
`TRIGger<m>:DPHY:ESDTa:MIN` on page 2253

Data Index
Data index sets the range within this series of the data value that is considered for the analysis. The data index setup consists of the condition and one or two data index values.

"Condition" Sets the comparison condition to a specific value or a range.

"Index Min/Index" Specifies the value or sets the start value of a range.

"Index Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
For "Type">"HS Data":
`TRIGger<m>:DPHY:DIDX:CONDition` on page 2250
`TRIGger<m>:DPHY:DIDX:MAX` on page 2250
`TRIGger<m>:DPHY:DIDX:MIN` on page 2251
For "Type">"LP Escape Mode":
`TRIGger<m>:DPHY:ESINdex:CONDition` on page 2253
`TRIGger<m>:DPHY:ESINdex:MAX` on page 2254
`TRIGger<m>:DPHY:ESINdex:MIN` on page 2254

Escape mode
The escape mode setup consists of the condition and one or two data patterns.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Condition" Sets the comparison condition to a specific value or a range.

"Data Min/Data" Specifies the value or sets the start value of a range.

"Data Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:DPHY:ESCMode:CONDition` on page 2252
`TRIGger<m>:DPHY:ESCMode:MAX` on page 2252
`TRIGger<m>:DPHY:ESCMode:MIN` on page 2252

13.23.3.2 Triggering on D-PHY

Prerequisite: A bus is configured for the D-PHY signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Tap "Serial bus" and select the serial bus that is set to D-PHY.
5. Select the "Trigger Type" to be used for D-PHY protocol analysis.
6. To refine the trigger settings, configure additional settings, which are available for some trigger types.
   For details, see Chapter 13.23.3.1, "D-PHY trigger settings", on page 879.

13.23.4 D-PHY decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Serial Bus" dialog > "Setup" tab, enable "Decode".
2. In the "Serial Bus" dialog > "Display" tab, select additional result display settings: "Show table" and "Show binary".
3. If required, enable "Zoom coupling"

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

If a frame is marked red, some kind of error occurred. To analyze the error, see the indication of the corresponding frame in the result table.

ECC checks are supported. ECC can correct a certain number of bit errors in the header. If the errors can be fixed, the status is an ECC warning. If there are too many errors, an ECC error is shown.

Decode Results Table

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

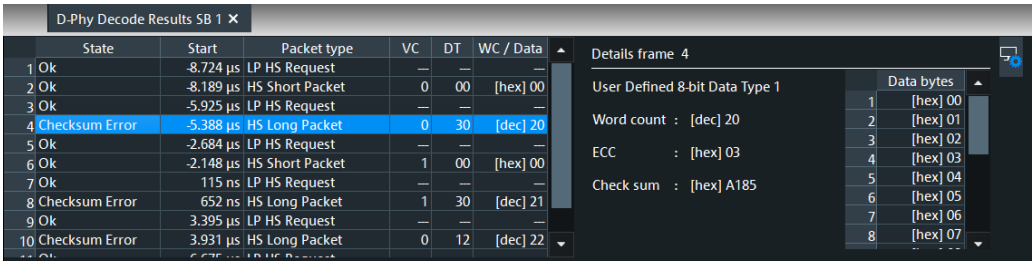


Table 13-45: Content of the Decode results table

Column	Description
State	Overall state of the frame
Start	Time of frame start in relation to the trigger point
Packet Type	Packet type
VC	Number of virtual channels
DT	Data type
WC/ Data	Word count or data value
Number of Packets	Number of packets in the frame

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands are described in Chapter 24.17.24.3, "D-PHY decode results", on page 2255.

13.23.5 Search on decoded D-PHY data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

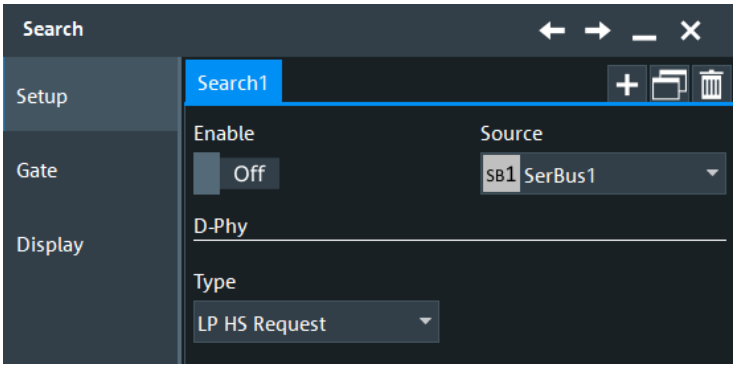
Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 11, "Search functions", on page 419.

13.23.5.1 D-PHY search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"



Type
Searches for the selected D-PHY type.

Remote command:
`SEARch:TRIGger:DPHY:TYPE` on page 2261

HS Start of Packet ← Type
Searches for a transmission start of a high-speed packet.

HS End of Packet ← Type
Searches for a transmission end of a high-speed packet.

HS Packet Header ← Type
Searches for a packet header of a high-speed package. The header consists of a data identifier (containing the Virtual Channel and the Data Type) and a Word Count/ Data. You can specify the values each part of the packet header.

HS Data ← Type
Searches for a specified high-speed data.

LP Escape Mode ← Type
Searches for an escape mode event.

LP Lane Turnaround ← Type
Searches for a low-power turnaround, a reversion of the transmission direction.

LP HS Request ← Type
Searches for a high-speed request.

Virtual Channel
Searches for a specific virtual channel, an independent data stream for up to four peripherals.

Remote command:
`SEARch:TRIGger:DPHY:HSVC` on page 2268

Data Type
Searches for a specific data type. The data type setup consists of the condition and one or two data patterns.

"Condition" Defines the operator to set a specific data type ("Equal" or "Not equal") or a data type range.

"Data Min/Data"
Defines the bit pattern of the data pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:DPHY:DTYPe:CONDition` on page 2264
`SEARch:TRIGger:DPHY:DTYPe:MAX` on page 2264
`SEARch:TRIGger:DPHY:DTYPe:MIN` on page 2264

Word Count/ Data
Searches for a specific word count /data. The word count setup consists of the condition and one or two data patterns.

"Condition" Defines the operator to set a specific word count ("Equal" or "Not equal") or a word count range.

"Data Min/Data"
Defines the bit pattern of the word pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second word pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:DPHY:WCOunt:CONDition` on page 2268
`SEARch:TRIGger:DPHY:WCOunt:MAX` on page 2268
`SEARch:TRIGger:DPHY:WCOunt:MIN` on page 2269

Data Value
Searches for a specific data value. The data value setup consists of the condition and one or two data value patterns.

"Condition" Defines the operator to set a specific data value ("Equal" or "Not equal") or a data value range.

"Data Min/Data"
Defines the bit pattern of the data value pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data value pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
For "Type">"HS Data":
`SEARch:TRIGger:DPHY:DATA:CONDition` on page 2262
`SEARch:TRIGger:DPHY:DATA:MAX` on page 2262
`SEARch:TRIGger:DPHY:DATA:MIN` on page 2262
For "Type">"LP Escape Mode":
`SEARch:TRIGger:DPHY:ESDTa:CONDition` on page 2266
`SEARch:TRIGger:DPHY:ESDTa:MAX` on page 2266
`SEARch:TRIGger:DPHY:ESDTa:MIN` on page 2266

Data Index
Data index sets the range within this series of the data value to be searched for. The data index setup consists of the condition and one or two data index values.

"Condition" Defines the operator to set a specific data ("Equal") or a data range.

"Index Min/Index"
Defines the minimum index.

"Index Max" The second index pattern is required to specify a range with conditions "In range".

Remote command:
For "Type">"HS Data":
`SEARch:TRIGger:DPHY:DIDX:CONDition` on page 2263
`SEARch:TRIGger:DPHY:DIDX:MAX` on page 2263
`SEARch:TRIGger:DPHY:DIDX:MIN` on page 2263
For "Type">"LP Escape Mode":
`SEARch:TRIGger:DPHY:ESINdex:CONDition` on page 2267
`SEARch:TRIGger:DPHY:ESINdex:MAX` on page 2267
`SEARch:TRIGger:DPHY:ESINdex:MIN` on page 2267

Escape mode
Searches for an escape mode event. The escape mode setup consists of the condition and one or two data patterns.

"Condition" Defines the operator to set a specific word count ("Equal" or "Not equal") or a word count range.

"Data Min/Data"
Defines the bit pattern of the data pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:DPHY:ESCMode:CONDition` on page 2265
`SEARch:TRIGger:DPHY:ESCMode:MAX` on page 2265
`SEARch:TRIGger:DPHY:ESCMode:MIN` on page 2265

13.23.5.2 D-PHY search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands:

- `SEARch:RESult:DPHY:FCOunt?` on page 2269
- `SEARch:RESult:DPHY:FRAMe<m>:CS?` on page 2269
- `SEARch:RESult:DPHY:FRAMe<m>:DATA?` on page 2270
- `SEARch:RESult:DPHY:FRAMe<m>:DTName?` on page 2270
- `SEARch:RESult:DPHY:FRAMe<m>:DTYPe?` on page 2270
- `SEARch:RESult:DPHY:FRAMe<m>:ECC?` on page 2271
- `SEARch:RESult:DPHY:FRAMe<m>:PACKet<n>:IDX?` on page 2271
- `SEARch:RESult:DPHY:FRAMe<m>:PACKet<n>:VALue?` on page 2271
- `SEARch:RESult:DPHY:FRAMe<m>:STARt?` on page 2272
- `SEARch:RESult:DPHY:FRAMe<m>:STATe?` on page 2272
- `SEARch:RESult:DPHY:FRAMe<m>:STOP?` on page 2272
- `SEARch:RESult:DPHY:FRAMe<m>:TYPE?` on page 2273
- `SEARch:RESult:DPHY:FRAMe<m>:VCHannel?` on page 2273

13.23.5.3 Searching D-PHY

Prerequisite: A serial bus is configured for the D-PHY signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Tap "Setup search".
3. If the dialog box does not contain a search entry, tap the "Add" icon to create one, as described in "To create a user-defined search" on page 434.

4. Tap "Source" and select the serial bus that is set to D-PHY (e.g. "SB1", unless already selected).
5. Specify search criteria according to Chapter 13.23.5.1, "D-PHY search setup", on page 887.
6. To acquire a waveform, press [Single].

   The R&S RTO6 performs a D-PHY decode according to the thresholds and protocol settings of the associated serial bus source.
7. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

   The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.24 M-PHY and USB SSIC (option R&S RTO6-K580 and K570)

The M-PHY® is a serial communication protocol. Its specifications were developed by the Mobile Industry Processor Interface (MIPI) alliance as standards for the communication of high-speed components, like smartphones and tablets, within mobile devices. For more details on the specifications, refer to http://mipi.org/specifications/physical-layer#M-PHY Specification.

This protocol is the foundation for several upper layer protocols, such as UniPro℠ (Unified Protocol), which manage complex data transfer functions. Each of these protocols is optimized for its particular purpose, such as data storage, data transfer, display, camera, memory sharing and radio interface.

When implemented together with the MIPI UniPro, defined as UniPort-M, they deliver high speed, low power, scalable and flexible data transport solutions.

The super speed inter-chip (SSIC) was developed by the USB standards committee based on USB3.1 G1 (SuperSpeed). It is developed to support the chip to chip communication of USB3 components. The lower communication layers are using M-PHY as the encoder platform and use a multilane distribution similar to UniPro. The higher communication layers (above 8b/10b) are almost identical to USB3.1 G1.

- M-PHY basic 892
- M-PHY configuration 893
- M-PHY trigger 896
- M-PHY decode results 911
- Search on decoded M-PHY data 912

13.24.1 M-PHY basic

This chapter provides an overview of the characteristics of the M-PHY and UniPro protocols.

13.24.1.1 Data transfer

M-PHY words are always 10 bits long (both data and control words). For UniPro, they have variable lengths that depend on the frame type. For example, control words have 8 bits, data words have 16 bits and reserved words have 3 bits.

To achieve power efficiency and high-performance transmissions, multiple power-saving states and recovery times are utilized. Scalability and flexibility are achieved through the various transmission speed ranges and rates.

Different modes of operations:

- Disable mode: lowest power mode entered into once the power supply is turned on
- Hibernate (Hibern8): ultra low-power state, which can be used without configuration loss
- High-speed mode (HS): supports three gears with predefined data rates. This mode is used during high-speed transmission for transporting large volume of data. This mode utilizes the power-saving Stall state to reduce power consumption while offering a fast state transition in the range of nanoseconds
- Low-power mode (PWM): supports seven gears with predefined frequency ranges. This mode is used during low-speed transmission for power conservation. This mode utilizes the power-saving Sleep state with a state transition time typically in the range of microseconds

The R&S RTO6 supports all operating speed modes: high-speed and low-power mode. The gear settings of these modes are auto detected by the instrument.

13.24.1.2 Data analysis

The M-PHY and UniPro decoding process involves several stages, similar to D-PHY.

The stages are as the following:

- Stage 1: involves two substages:
  - Stage 1a: converts the three stage cross points into bit and attempts a HS-Sync. This stage involves the Hibern8-Detection/Filter, NRZ Unclocked Decoder and Burst Detection.
  - Stage 1b is only involved if Stage 1a burst detection fails, assuming using PWM. Input is from the bits from stage 1a. This stage involves the PWM Decoder, Line CFG Decoder and Burst Detection.
- Stage 2: performs the 8b/10b decoding of bits from stage 1 into Bytes and identifies the 8b/10b control words. This stage involves the 8b/10b Decoder and Shift-decoder to Bytes.
- Stage 3: starts decoding the Bytes from stage 2. This stage involves the Descrambler and 17-bit Shift Decoder.

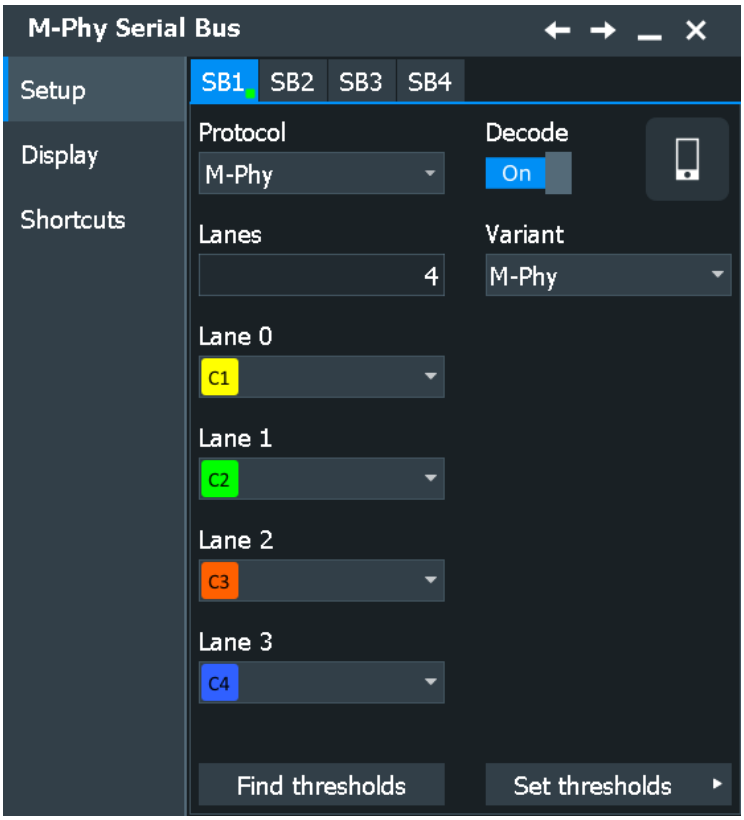
- Stage 4: merges the data lanes and finalizes the decoding to the UniPro data frames. This stage involves the Lane Merger, PDU Sync Detector, Shift Decode and Dynamic Data unit Decoder.

13.24.2 M-PHY configuration

13.24.2.1 M-PHY configuration settings

Access: [Bus] > "M-PHY" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Variant
Selects the protocol running on the interface. You can select between the M-PHY, UniPro and SSIC.

Remote command:
`BUS<m>:MPHY:VARiant` on page 2274

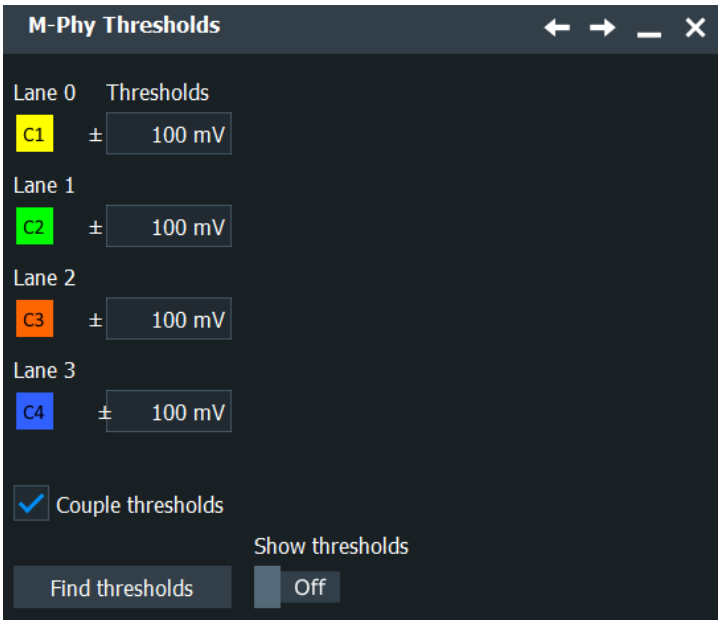
Lanes
Sets the number of logical lanes that are mapped to the physical lines. You can select up to four lanes.

Remote command:
`BUS<m>:MPHY:DLANes` on page 2274

Lane 0 / Lane 1 / Lane 2 / Lane 3: Channel
Select the signal sources for the logical lanes.

Remote command:
`BUS<m>:MPHY:DZERo:SOURce` on page 2275
`BUS<m>:MPHY:DONE:SOURce` on page 2275
`BUS<m>:MPHY:DTWO:SOURce` on page 2275
`BUS<m>:MPHY:DTHRee:SOURce` on page 2275

Threshold setup
Opens a dialog, where you can set the tresholds for digitization of signals for each lane.



"Thresholds"	Enter individual values for each line directly in the fields.
"Couple thresholds"	Enables the same threshold value for all lanes.
"Find thresholds"	Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.
"Show thresholds"	If enabled, the threshold levels are displayed in the diagram.

Remote command:
`BUS<m>:MPHY:DZERo:THReshold` on page 2275
`BUS<m>:MPHY:DONE:THReshold` on page 2275
`BUS<m>:MPHY:DTWO:THReshold` on page 2275

BUS<m>:MPHY:DTHRee:THReshold on page 2275
BUS<m>:MPHY:THCoupling on page 2276

Scrambled Mode
Available, if "Variant" = "SSIC".

Selects, if the SSIC data is scrambled or descrambled. Set this parameter according to your signal parameters, to ensure a proper decoding.

Remote command:
BUS<m>:MPHY:SCRMode on page 2276

13.24.2.2 Display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

To enhance the decode possibilities of the M-PHY or UniPro protocol, you can use an additional setting in the "Display" tab: "Decode layer".

Decode layer
Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

"Final" Final decoded frames of each lane.

"Edges" All edges of each lane in stage 1 decoding.

"Bits" All bits of each lane in stage 1 decoding.

"8b/10b Symbols"
8b/10b symbols in stage 2 decoding.

"LCC bits" LCC bits in stage 3 decoding.

"Filter/descrambler"
UniPro filter/descrambler in stage 3 decoding.

"Lane merge"
UniPro lane merge in stage 4 decoding.

"Bytes" UniPro bytes in stage 2 decoding.

Show multiple lanes
Enables the display of the results in multiple lanes.

Remote command:
BUS<m>:MPHY:SMLanes on page 2276

13.24.2.3 Configuring the M-PHY signals

Access: [Bus] > "Setup" tab > "Protocol" = *M-PHY*

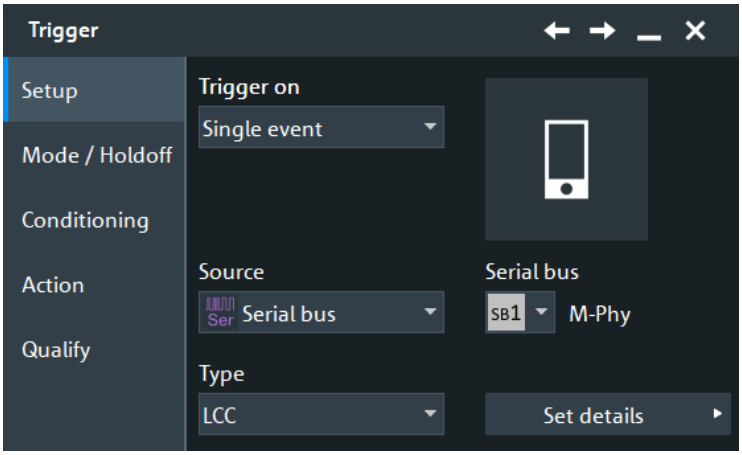
Assign the lanes to the channels and define the thresholds:

1. Press [Bus] on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. In the "Setup" tab, select the protocol: M-PHY.
4. Tap "Variant" and select the protocol.
5. Enter the number of "Lanes".
6. Select the source "Channel" for each lane.
7. Enter the "Thresholds" for each lane.
8. Enable "Decode".

13.24.3 M-PHY trigger

13.24.3.1 M-PHY trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Type
Selects the trigger type for the analysis of the selected protocol.

Remote command:
`TRIGger<m>:MPHY:TYPE` on page 2278

Start of frame ← Type
Triggers on the start of an M-PHY or a UniPro frame.

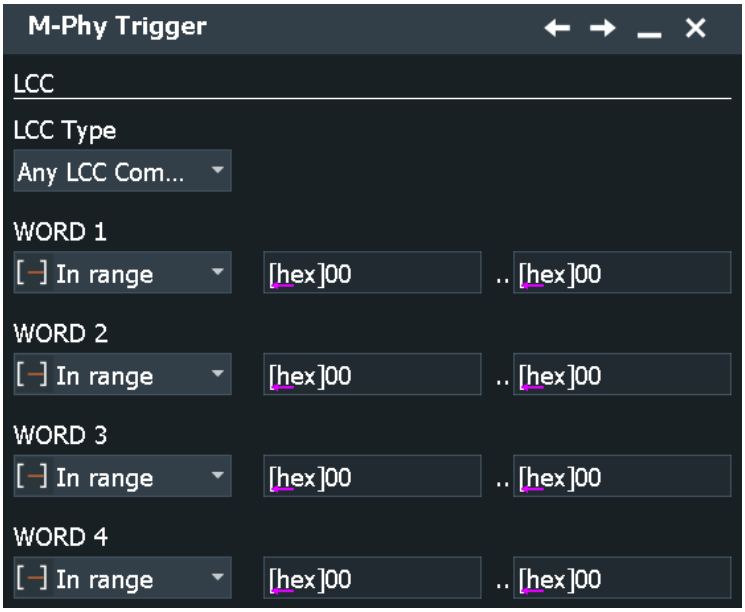
Burst ← Type
Triggers on an M-PHY burst frame.

Adapt ← Type
Triggers on an M-PHY Adapt frame. This control frame is used for clock/bit synchronization.

LCC ← Type
Triggers on an M-PHY Line Control Command (LCC) frame that matches the specified:

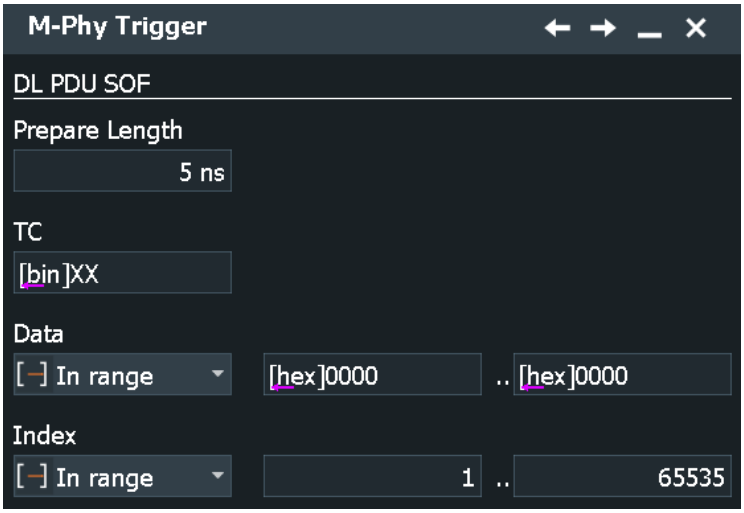
- LCC Type
- WORD

This control frame is used for configuring the line to a different state or mode depending on the LCC type.



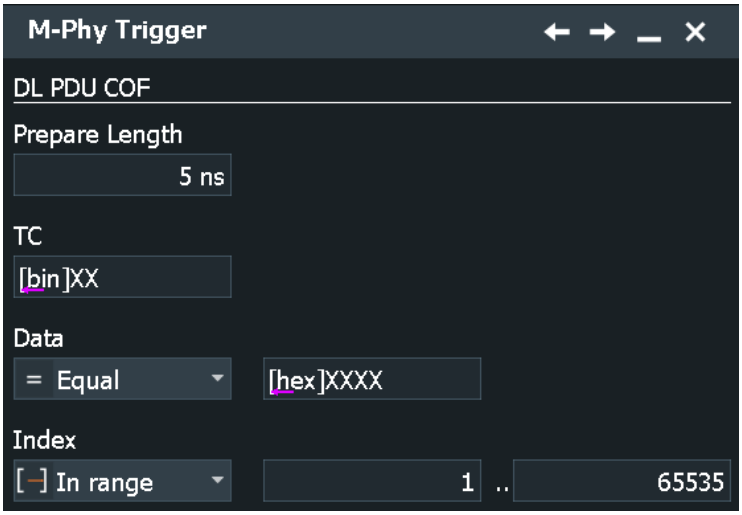
DL PDU SOF ← Type
Triggers on a UniPro Downlink Protocol Data Unit (DL PDU) Start Of Frame (SOF) that matches the specified:

- Prepare length
- TC
- Data
- Index



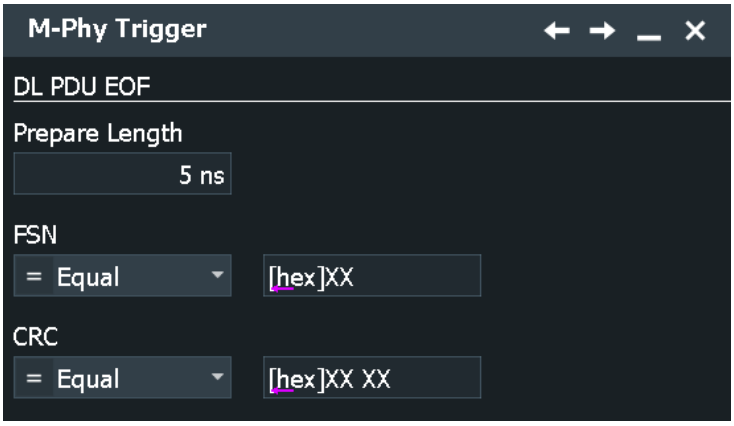
DL PDU COF ← Type
Triggers on a UniPro DL PDU Continuation Of Pre-empted Frame (COF) that matches the specified:

- Prepare length
- TC
- Data
- Index



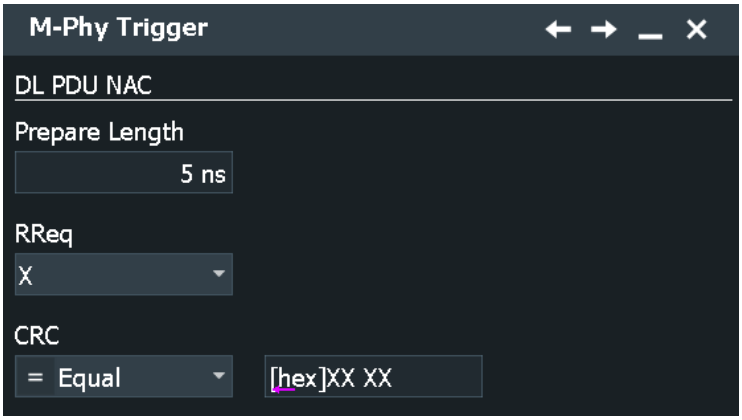
DL PDU EOF ← Type
Triggers on a UniPro DL PDU End Of Frame (EOF) frame that matches the specified conditions:

- Prepare length
- FSN
- CRC



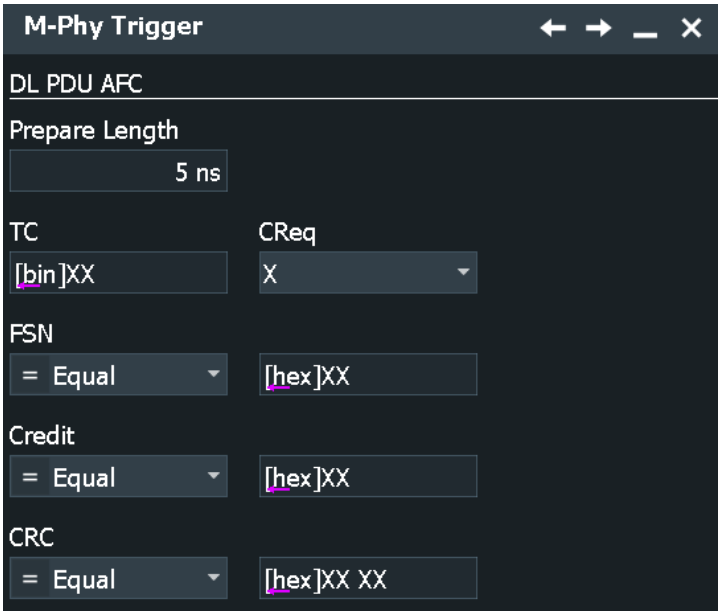
DL PDU NAC ← Type
Triggers on a UniPro DL PDU Negative Acknowledgment Control (NAC) frame that matches the specified conditions:

- Prepare length
- RReq
- CRC



DL PDU AFC ← Type
Triggers on a UniPro DL PDU Acknowledgement and L2 Flow Control (AFC) frame that matches the specified conditions:

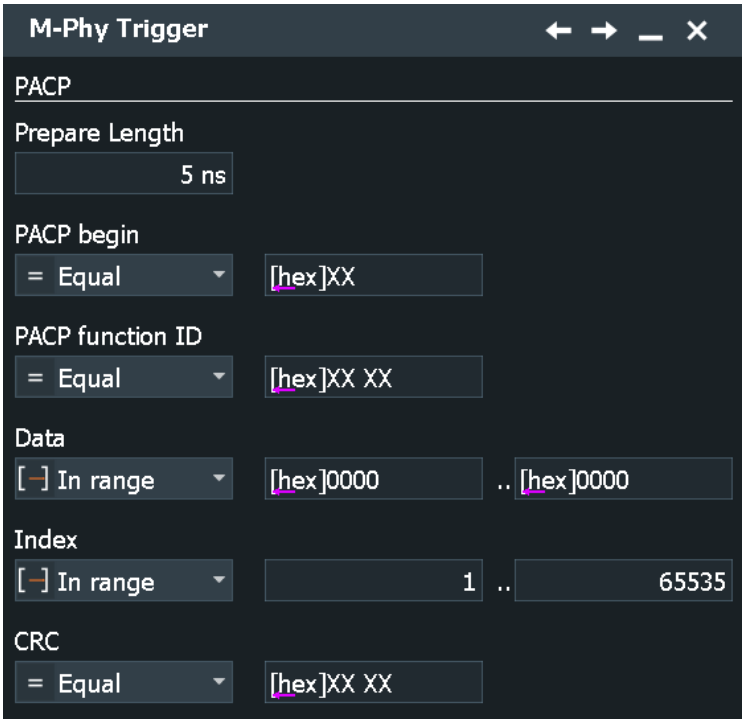
- Prepare length
- TC
- CReq
- FSN
- Credit
- CRC



PACP ← Type
Triggers on a UniPro PHY Adapter Control Protocol (PACP) frame that matches the specified condition:

- Prepare length
- PACP Begin
- PACP Function ID
- Data
- Index
- CRC

This control frame is used mainly for power mode change and L1.5 link management. It is also used in scrambling request.

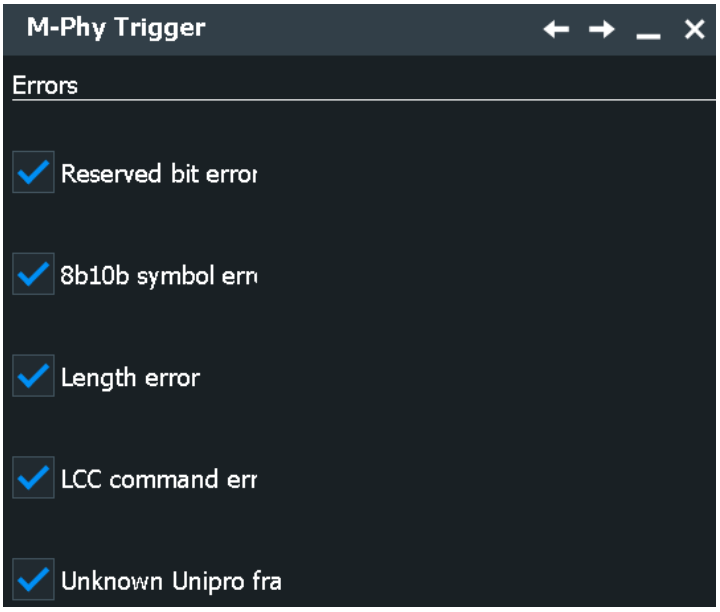


Trigger Upr0 ← Type
Triggers on a UniPro Trigger Upper0 frame. This control frame is used for link startup sequence.

Trigger Upr1 ← Type
Triggers on a UniPro Trigger Upper1 frame. This control frame is used for link startup sequence.

Trigger Upr2 ← Type
Triggers on a UniPro Trigger Upper2 frame. This control frame is used for link startup sequence.

Errors ← Type
Triggers on the specified M-PHY or UniPro error frame.



Prepare length
Sets the prepare length.

LCC Type
Selects the type of LCC command to be triggered on.

"Any LCC Command"
All the available LCC commands

"HIBERN8-SLEEP"
Switches the power-saving state to ultra-low power without configuration loss. This state saves up to 90% power and is used with PWM burst.

"HIBERN8-STALL"
Switches the power-saving state to ultra-low power without configuration loss. This state saves up to 75% power and is used with HS burst.

"READ-CAPABILITY"
Recovers data about the Optical Media Converter (OMC) capabilities

"READ-MFG-INFO"
Retrieves manufacturing ID and vendor-specific information

"READ-VEND-INFO"
Retrieves the additional four delimited bytes containing vendor-specific information

"WRITE-ATTRIBUTE"
Sets the configuration parameters required for lane operation

"PWM-G0/PWM-G1/PWM-G2/PWM-G3/PWM-G4/PWM-G5/PWM-G6/PWM-G7"
Switches the transmission mode to the selected low-power gear

"HS-G1A/HS-G2A/HS-G3A/HS-G4A/HS-G1B/HS-G2B/HS-G3B/HS-G4B"
Switches the transmission mode to the selected high-speed gear

"Reserved"
Reserved bit command. These bits are reserved for future use.

Remote command:
`TRIGger<m>:MPHY:LCCType` on page 2279

WORD 1/WORD 2/WORD 3/WORD 4
Sets the specified words to be triggered on. The setup for a word consists of the condition and one or two word patterns.

"Condition" Sets the comparison condition to a specific value or a range.

"Data Min/Data"
Specifies the value or sets the start value of a range.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
For " WORD 1":
`TRIGger<m>:MPHY:LWONe:CONDition` on page 2279
`TRIGger<m>:MPHY:LWONe:MIN` on page 2280
`TRIGger<m>:MPHY:LWONe:MAX` on page 2280
For " WORD 2":
`TRIGger<m>:MPHY:LWTWo:CONDition` on page 2280
`TRIGger<m>:MPHY:LWTWo:MIN` on page 2281
`TRIGger<m>:MPHY:LWTWo:MAX` on page 2281
For " WORD 3":
`TRIGger<m>:MPHY:LWTHree:CONDition` on page 2281
`TRIGger<m>:MPHY:LWTHree:MIN` on page 2282
`TRIGger<m>:MPHY:LWTHree:MAX` on page 2282
For " WORD 4":
`TRIGger<m>:MPHY:LWFour:CONDition` on page 2282
`TRIGger<m>:MPHY:LWFour:MIN` on page 2283
`TRIGger<m>:MPHY:LWFour:MAX` on page 2283

TC
Sets the specified Traffic Class (TC) to be triggered on.

TC is used for classifying the traffic flow based on protocol and port number, e.g. TC0 and TC1 are two priority classes with guaranteed link reliability defined and used in L2.

"Data" Defines the bit pattern of the TC pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

Remote command:
`TRIGger<m>:MPHY:TC` on page 2283

Data

Sets the data to be triggered on. The trigger data setup consists of a comparison condition and one or two data patterns.

"Condition" Sets the comparison condition to a specific value or a range.

"Data Min/Data"
Specifies the value or sets the start value of a range.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:MPHY:DATA:DCON` on page 2284
`TRIGger<m>:MPHY:DATA:DMIN` on page 2284
`TRIGger<m>:MPHY:DATA:DMAX` on page 2284

Index

The trigger index setup consists of a comparison condition and one or two address patterns.

"Condition" Sets the comparison condition to a specific value or a range.

"Index Min/Index"
Specifies the value or sets the start value of a range.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Index Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:MPHY:DATA:ICONdition` on page 2285
`TRIGger<m>:MPHY:DATA:IMIN` on page 2285
`TRIGger<m>:MPHY:DATA:IMAX` on page 2285

FSN

Sets the specified Frame Sequence Number (FSN) to be triggered on. The FSN setup consists of the condition and one or two FSN patterns.

"Condition" Defines the operator to set a specific FSN, e.g. "Equal" or "Not Equal" or a range.

"Data Min/Data"
Defines the bit pattern of the FSN pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second FSN pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:MPHY:FSNumber:CONDition` on page 2285
`TRIGger<m>:MPHY:FSNumber:MIN` on page 2286
`TRIGger<m>:MPHY:FSNumber:MAX` on page 2286

CRC
Sets the specified Cyclic Redundancy Check (CRC) to be triggered on. The CRC setup consists of the condition and one or two CRC patterns.

CRC is an error detecting code to detect accidental changes to raw data.

"Condition" Defines the operator to set a specific CRC, e.g. "Equal" or "Not Equal" or a range.

"Data Min/Data"
Defines the bit pattern of the CRC pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`TRIGger<m>:MPHY:CRC:CONDition` on page 2286
`TRIGger<m>:MPHY:CRC:MIN` on page 2287
`TRIGger<m>:MPHY:CRC:MAX` on page 2287

Req
Sets the specified Reset Link Request (RReq) or Credit Transmit Request (CReq) to be triggered on.

RReq is used for requesting the remote end to reinitialize its Transmit (TX) Physical Layer (PHY) while CReq is used for requesting flow control information for the corresponding TC from the remote end.

"Data" Defines the bit pattern of the RReq pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

Remote command:
`TRIGger<m>:MPHY:RREQ` on page 2288
`TRIGger<m>:MPHY:CREQ` on page 2287

Credit
Sets the specified credit to be triggered on. The credit setup consists of the condition and one or two credit patterns.

Credit represents the total number of credits available since boot time.

"Condition" Defines the operator to set a specific credit, e.g. "Equal" or "Not Equal" or a range.

"Data Min/Data"
Defines the bit pattern of the credit pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second credit pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:MPHY:CREDit:CONDition` on page 2288
`TRIGger<m>:MPHY:CREDit:MIN` on page 2289
`TRIGger<m>:MPHY:CREDit:MAX` on page 2289

PACP Begin
Sets the specified PACP Begin to be triggered on. The PACP Begin setup consists of the condition and one or two PACP Begin patterns.

"Condition" Defines the operator to set a specific PACP Begin, e.g. "Equal" or "Not Equal" or a range.

"Data Min/Data"
Defines the bit pattern of the data value pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second PACP Begin pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:MPHY:PBEGin:CONDition` on page 2289
`TRIGger<m>:MPHY:PBEGin:MIN` on page 2289
`TRIGger<m>:MPHY:PBEGin:MAX` on page 2290

PACP Function ID
Sets the specified PACP Function ID to be triggered on. The PACP Function ID setup consists of the condition and one or two PACP Function ID patterns.

"Condition" Defines the operator to set a specific PACP Function ID, e.g. "Equal" or "Not Equal" or a range.

"Data Min/Data"
Defines the bit pattern of the PACP Function ID pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second PACP Function ID pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`TRIGger<m>:MPHY:PFID:CONDition` on page 2290
`TRIGger<m>:MPHY:PFID:MIN` on page 2290
`TRIGger<m>:MPHY:PFID:MAX` on page 2291

Errors
Sets the type of error events to be triggered on.

Reserved Bit Error ← Errors
Triggers on reserved bit error.

This error means that the waveform violates the reserved field according to the specification. For example, a DL_PDU_AFC frame requires that the two bits before and three bits after the FSN field to be reserved and set as 0. If the waveform carries 1 in these positions instead, it is marked as this error.

Remote command:
`TRIGger<m>:MPHY:ERRor:REServed` on page 2292

LCC Command Error ← Errors
Triggers on LCC command error.

This error is marked when the Command field in the LCC packet is not a known command defined in the specification.

Remote command:
`TRIGger<m>:MPHY:ERRor:LCMD` on page 2291

8b10b Symbol Error ← Errors
Triggers on 8b10b symbol error.

For example, some of the 10-bit combinations that do not map to 8-bit according to the specification are invalid 10-bit and therefore marked as this error.

Remote command:
`TRIGger<m>:MPHY:ERRor:SYMBol` on page 2292

Unknown Unipro Frame ← Errors
Triggers on unidentified UniPro frame error. This error is marked when the end mark on UniPro packets is undetected.

Remote command:
`TRIGger<m>:MPHY:ERRor:UNKNown` on page 2292

Length Error ← Errors
Triggers on frames that violate the length according to the specification.

Remote command:
`TRIGger<m>:MPHY:ERRor:LENGth` on page 2291

13.24.3.2 Triggering on M-PHY

Prerequisite: A bus is configured for the M-PHY signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [Bus] key and select the "Trigger" tab.
2. Press the "Select" button. The "Source" button is shown.
3. Tap the "Source" button and select the "Serial bus" trigger source.

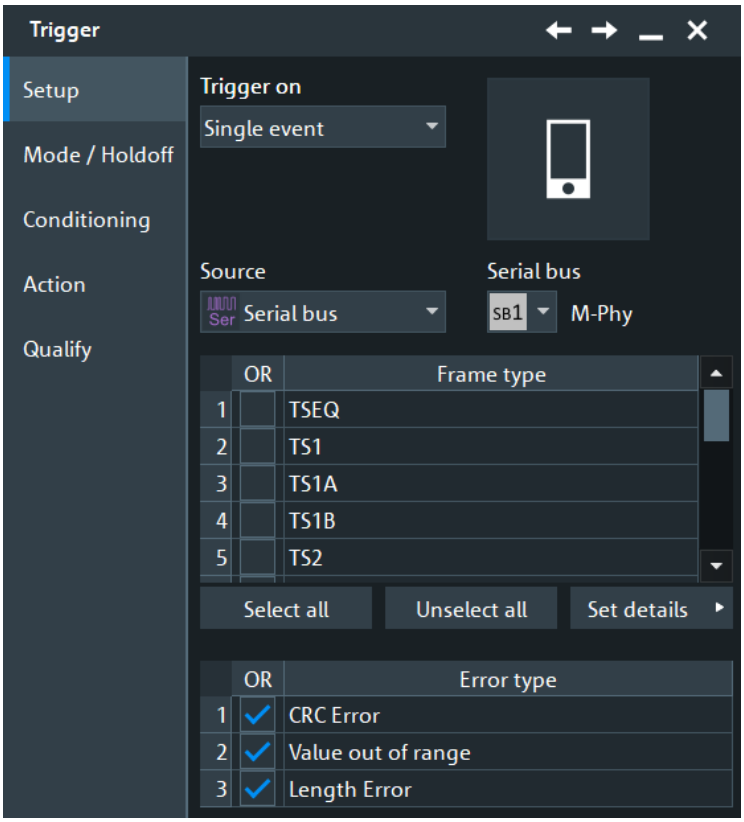
4. Tap "Serial bus" and select the serial bus that is set to M-PHY, e.g.:

The "Protocol" selection is then automatically set to "M-PHY".

5. Select the "Type" to use in the M-PHY protocol analysis.
6. To refine the trigger settings, configure additional settings, which are available for some trigger types.
For details, see Chapter 13.24.3.1, "M-PHY trigger settings", on page 896.

13.24.3.3 SSIC trigger settings

The setting is available for SSIC option R&S RTO6-K570.



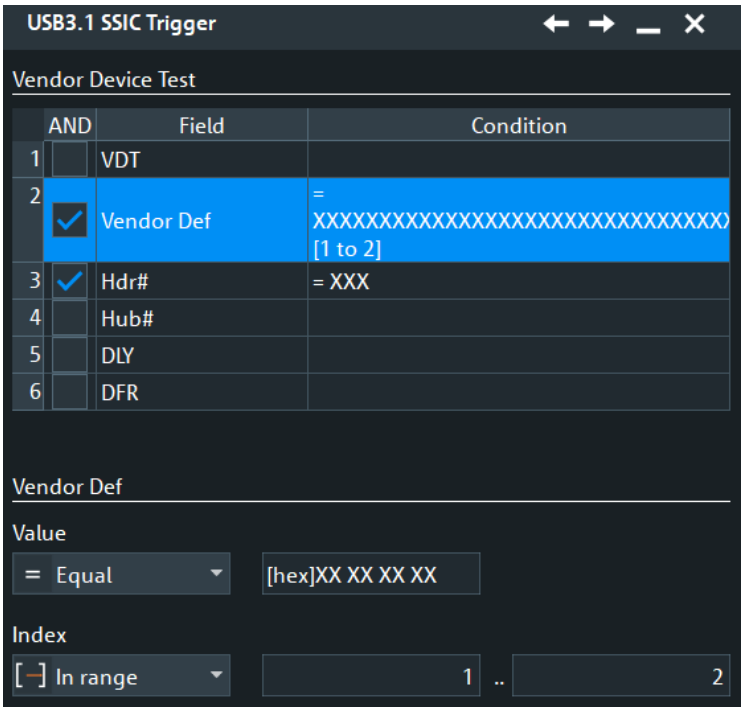
Frame type
Selects the frame type for the M-PHY SSIC.

The SSIC trigger frames and fields are the same as for USB3.1, see "Frame types" on page 842.

Error types, Enable
The table lists the error types you can trigger on. Select the required errors in the "OR" column.

Remote command:
`TRIGger<m>:MPHY:SSIC:ERENable` on page 2321
`TRIGger<m>:MPHY:SSIC:ERRor<n>:ENABle` on page 2321

Set details
Opens a dialog to define frame specific settings.



AND / Field name / Condition ← Set details
This table lists the field numbers and names in the selected frame together with a summary of the user settings of checking conditions for each field and a checkbox to enable the checking.

Select a field in the table to specify the checking conditions for this field in the "Set details" dialog.

The SSIC trigger frames and fields are the same as for USB3.1, see "Frame types" on page 842.

Remote command:
`TRIGger<m>:MPHY:SSIC:FRENable` on page 2321
`TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:ENABle` on page 2322

Data
Defines for the selected field, how a data check is executed.

"Condition"	Defining specific data or a data range requires to set the operator to one of the following conditions: equal, not equal, less than, less than or equal, greater than, greater than or equal, in range, out of range.
"Min"	Specifies data or sets the start value of a data range.
"Max"	Sets the end value of a data range, if "Condition" is set to "In range" or "Out of range".

Remote command:
`TRIGger<m>:MPHY:SSIC:DOPerator` on page 2323
`TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DOPerator` on page 2323
`TRIGger<m>:MPHY:SSIC:DMIN` on page 2323
`TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DMIN` on page 2323
`TRIGger<m>:MPHY:SSIC:DMAX` on page 2323
`TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DMAX` on page 2323

BitState
Defines the bit state to be checked for the selected field. Permitted bit states are "1", "0" or "X" (do not care).

Remote command:
`TRIGger<m>:MPHY:SSIC:BIT` on page 2322
`TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:BIT` on page 2322

Index
Defines for the selected field, how an index check is executed.

"Condition"	Defining a specific index or an index range requires to set the operator to one of the following conditions: equal, in range.
"Min"	Specifies the index or sets the start value of an index range.
"Max"	Sets the end value of an index range, if "Condition" is set to "In range".

Remote command:
`TRIGger<m>:MPHY:SSIC:IOPerator` on page 2325
`TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IOPerator` on page 2325
`TRIGger<m>:MPHY:SSIC:IMIN` on page 2324
`TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IMIN` on page 2324
`TRIGger<m>:MPHY:SSIC:IMAX` on page 2324
`TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IMAX` on page 2324

13.24.3.4 Triggering on SSIC

Prerequisite: A bus is configured for the "M-PHY" > "Variant" = "SSIC" signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Tap "Serial bus" and select the serial bus that is set to SSIC.
5. If you trigger on errors, select the error types you want to find.
6. Select the frame type to be used for SSIC protocol analysis.
7. For some frame types, you can define the frame fields.

For details, see Chapter 13.24.3.3, "SSIC trigger settings", on page 908.

13.24.4 M-PHY decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".

2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

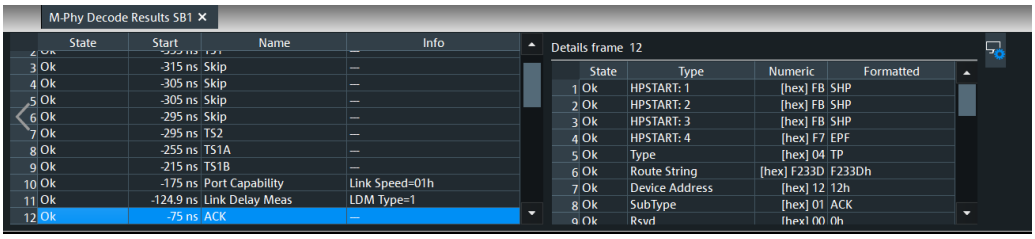


Table 13-46: Content of the Decode results table

Column	Description
Frame	Frame count
State	Overall state of the frame indicating, for example, if the frame is valid or invalid
Start	Time of frame start in relation to the trigger point
Info	Label on top of the frame

Table 13-47: Content of the Details frame table

Column	Description
State	State of the field
Type	Frame identifier specifying the data or control frame name

Column	Description
Numeric	Hexadecimal value of the field
Formatted	Formatted content of the field

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands are described in Chapter 24.17.25.3, "Decode results M-PHY", on page 2292.

13.24.5 Search on decoded M-PHY data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

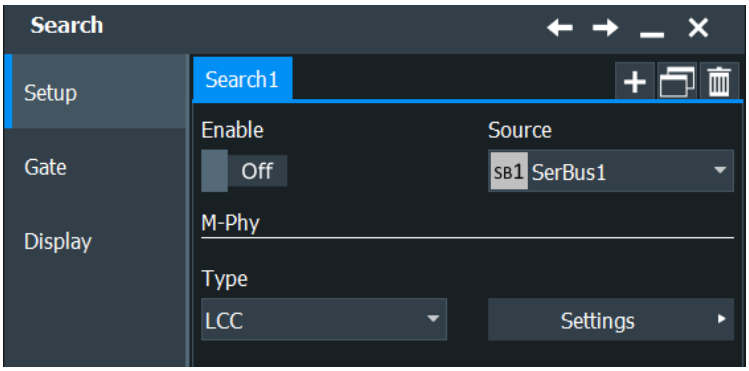
Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 11, "Search functions", on page 419.

13.24.5.1 M-PHY search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"



Type
Selects the search type for the selected protocol.

Remote command:
`SEARch:TRIGger:MPHY:TYPE` on page 2300

Start of frame ← Type
Searches for the start of an M-PHY or a UniPro frame.

Burst ← Type
Searches for an M-PHY burst.

Adapt ← Type
Searches for an M-PHY Adapt frame. This control frame is used for clock/bit synchronization.

LCC ← Type
Searches for an M-PHY Line Control Command (LCC) frame that matches the specified LCC Type or any WORD condition.

This control frame is used for configuring the line to a different state or mode depending on the LCC type.

DL PDU SOF ← Type
Searches for a UniPro Downlink Protocol Data Unit (DL PDU) Start Of Frame (SOF) that matches the specified TC, Data or Index condition.

DL PDU COF ← Type
Searches for a UniPro DL PDU Continuation Of Pre-empted Frame (COF) that matches the specified TC, Data or Index condition.

DL PDU EOF ← Type
Searches for a UniPro DL PDU End Of Frame (EOF) that matches the specified FSN or CRC condition.

DL PDU NAC ← Type
Searches for a UniPro DL PDU Negative Acknowledgment Control (NAC) frame that matches the specified RReq or CRC condition.

DL PDU AFC ← Type
Searches for a UniPro DL PDU Acknowledgement and L2 Flow Control (AFC) frame that matches the specified TC, CReq, FSN, Credit or CRC condition.

PACP ← Type
Searches for a UniPro PHY Adapter Control Protocol (PACP) frame that matches the specified PACP Begin, PACP Function ID, Data, Index or CRC condition.

This control frame is used mainly for power mode change and L1.5 link management. It is also used in scrambling request.

Trigger Upper0 ← Type
Searches for a Trigger Upper0 frame. This control frame is used for link startup sequence.

Trigger Upper1 ← Type
Searches for a Trigger Upper1 frame. This control frame is used for link startup sequence.

Trigger Upper2 ← Type
Searches for a Trigger Upper2 frame. This control frame is used for link startup sequence.

Errors ← Type
Searches for the specified error frame.

LCC Type
Selects the type of LCC command to be searched for.

"Any LCC Command"
All the available LCC commands

"HIBERN8-SLEEP"
Switches the power-saving state to ultra-low power without configuration loss. This state saves up to 90% power and is used with PWM burst.

"HIBERN8-STALL"
Switches the power-saving state to ultra-low power without configuration loss. This state saves up to 75% power and is used with HS burst.

"READ-CAPABILITY"
Recovers data about the Optical Media Converter (OMC) capabilities

"READ-MFG-INFO"
Retrieves manufacturing ID and vendor-specific information

"READ-VEND-INFO"
Retrieves the additional four delimited bytes containing vendor-specific information

"WRITE-ATTRIBUTE"
Sets the configuration parameters required for lane operation

"PWM-G0/PWM-G1/PWM-G2/PWM-G3/PWM-G4/PWM-G5/PWM-G6/PWM-G7"
Switches the transmission mode to the selected low-power gear

"HS-G1A/HS-G2A/HS-G3A/HS-G4A/HS-G1B/HS-G2B/HS-G3B/HS-G4B"
Switches the transmission mode to the selected high-speed gear

"Reserved"
Reserved bit command. These bits are reserved for future use.

Remote command:
`SEARch:TRIGger:MPHY:LCCType` on page 2301

WORD 1/WORD 2/WORD 3/WORD 4
Sets the specified words to be searched for. The setup for a word consists of the condition and one or two word patterns.

"Condition" Defines the operator to set a specific word, e.g. "Equal" or " Not Equal" or a range.

"Data Min/Data"
Defines the bit pattern of the word pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second word pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
For " WORD 1":
`SEARch:TRIGger:MPHY:LWONe:CONDition` on page 2302
`SEARch:TRIGger:MPHY:LWONe:MIN` on page 2302
`SEARch:TRIGger:MPHY:LWONe:MAX` on page 2302
For " WORD 2":
`SEARch:TRIGger:MPHY:LWTWo:CONDition` on page 2303
`SEARch:TRIGger:MPHY:LWTWo:MIN` on page 2303
`SEARch:TRIGger:MPHY:LWTWo:MAX` on page 2303
For " WORD 3":
`SEARch:TRIGger:MPHY:LWTHree:CONDition` on page 2304
`SEARch:TRIGger:MPHY:LWTHree:MIN` on page 2304
`SEARch:TRIGger:MPHY:LWTHree:MAX` on page 2304
For " WORD 4":
`SEARch:TRIGger:MPHY:LWFour:CONDition` on page 2305
`SEARch:TRIGger:MPHY:LWFour:MIN` on page 2305
`SEARch:TRIGger:MPHY:LWFour:MAX` on page 2305

TC
Sets the specified Traffic Class (TC) to be searched for.

TC is used for classifying the traffic flow based on protocol and port number, e.g. TC0 and TC1 are two priority classes with guaranteed link reliability defined and used in L2.

"Data" Defines the bit pattern of the TC pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

Remote command:
`SEARch:TRIGger:MPHY:TC` on page 2306

Data
Sets the specified data to be searched for. The data setup consists of the condition and one or two data patterns.

"Condition" Defines the operator to set a specific data, e.g. "Equal" or "Not Equal") or a range.

"Data Min/Data"
Defines the bit pattern of the data pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second data pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:MPHY:DATA:DCONdition` on page 2306
`SEARch:TRIGger:MPHY:DATA:DMIN` on page 2306
`SEARch:TRIGger:MPHY:DATA:DMAX` on page 2307

Index
Index sets the range within this series of data that is considered for the search. The index setup consists of the condition and one or two data index values.

"Condition" Defines the operator to set a specific index, e.g. "Equal" or a range.

"Index Min/Index"
Defines the bit pattern of the index pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Index Max" The second index pattern is required to specify a range with conditions "In range".

Remote command:
`SEARch:TRIGger:MPHY:DATA:ICONdition` on page 2307
`SEARch:TRIGger:MPHY:DATA:IMIN` on page 2307
`SEARch:TRIGger:MPHY:DATA:IMAX` on page 2308

FSN
Sets the specified Frame Sequence Number (FSN) to be searched for. The FSN setup consists of the condition and one or two FSN patterns.

"Condition" Defines the operator to set a specific FSN, e.g. "Equal" or "Not Equal" or a range.

"Data Min/Data"
Defines the bit pattern of the FSN pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second FSN pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:MPHY:FSNumber:CONDition` on page 2308
`SEARch:TRIGger:MPHY:FSNumber:MIN` on page 2308
`SEARch:TRIGger:MPHY:FSNumber:MAX` on page 2309

CRC
Sets the specified Cyclic Redundancy Check (CRC) to be searched for. The CRC setup consists of the condition and one or two CRC patterns.

CRC is an error detecting code to detect accidental changes to raw data.

"Condition" Defines the operator to set a specific CRC, e.g. "Equal" or "Not Equal" or a range.

"Data Min/Data"
Defines the bit pattern of the CRC pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second CRC pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:MPHY:CRC:CONDition` on page 2309
`SEARch:TRIGger:MPHY:CRC:MIN` on page 2309
`SEARch:TRIGger:MPHY:CRC:MAX` on page 2310

Req
Sets the specified Reset Link Request (RReq) or Credit Transmit Request (CReq) to be searched for.

RReq is used for requesting the remote end to reinitialize its Transmit (TX) Physical Layer (PHY) while CReq is used for requesting flow control information for the corresponding `TC` from the remote end.

"Data" Defines the bit pattern of the RReq pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

Remote command:
For "RReq":
`SEARch:TRIGger:MPHY:RREQ` on page 2310
For "CReq"
`SEARch:TRIGger:MPHY:CREQ` on page 2310

Credit
Sets the specified credit to be searched for. The credit setup consists of the condition and one or two credit patterns.

Credit represents the total number of credits available since boot time.

"Condition" Defines the operator to set a specific credit, e.g. "Equal" or "Not Equal" or a range.

"Data Min/Data"
Defines the bit pattern of the credit pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second credit pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:MPHY:CREDit:CONDition` on page 2311
`SEARch:TRIGger:MPHY:CREDit:MIN` on page 2311
`SEARch:TRIGger:MPHY:CREDit:MAX` on page 2311

PACP Begin
Sets the specified PACP Begin to be searched for. The PACP Begin setup consists of the condition and one or two PACP Begin patterns.

"Condition" Defines the operator to set a specific PACP Begin, e.g. "Equal" or "Not Equal" or a range.

"Data Min/Data"
Defines the bit pattern of the data value pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second PACP Begin pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:MPHY:PBEGin:CONDition` on page 2312
`SEARch:TRIGger:MPHY:PBEGin:MIN` on page 2312
`SEARch:TRIGger:MPHY:PBEGin:MAX` on page 2312

PACP Function ID
Sets the specified PACP Function ID to be searched for. The PACP Function ID setup consists of the condition and one or two PACP Function ID patterns.

"Condition" Defines the operator to set a specific PACP Function ID, e.g. "Equal" or "Not Equal" or a range.

"Data Min/Data"
Defines the bit pattern of the PACP Function ID pattern.

In binary format, use the following characters: 1; 0; or X (do not care). To enter the pattern in other formats use the bit pattern editor, see Chapter 13.1.7, "Bit pattern editor", on page 493.

"Data Max" The second PACP Function ID pattern is required to specify a range with conditions "In range" and "Out of range".

Remote command:
`SEARch:TRIGger:MPHY:PFID:CONDition` on page 2313
`SEARch:TRIGger:MPHY:PFID:MIN` on page 2313
`SEARch:TRIGger:MPHY:PFID:MAX` on page 2313

Errors
Sets the type of error events to be searched for.

Reserved Bit Error ← Errors
Searches for reserved bit error.

This error means that the waveform violates the reserved field according to the specification. For example, a DL_PDU_AFC frame requires that the two bits before and three bits after the FSN field to be reserved and set as 0. If the waveform carries 1 in these positions instead, it is marked as this error.

Remote command:
`SEARch:TRIGger:MPHY:ERRor:REServed` on page 2314

LCC Command Error ← Errors
Searches for LCC command error.

This error is marked when the Command field in the LCC packet is not a known command defined in the specification.

Remote command:
`SEARch:TRIGger:MPHY:ERRor:LCMD` on page 2314

8b10b Symbol Error ← Errors
Searches for 8b10b symbol error.

For example, some of the 10-bit combinations that do not map to 8-bit according to the specification are invalid 10-bit and therefore marked as this error.

Remote command:
`SEARch:TRIGger:MPHY:ERRor:SYMBol` on page 2314

Unknown Unipro Frame ← Errors
Searches for unidentified UniPro frame error. This error is marked when the end mark on Unipro packets is undetected.

Remote command:
`SEARch:TRIGger:MPHY:ERRor:UNKNown` on page 2314

Length Error ← Errors
Searches for frames that violate the length according to the specification.

Remote command:
`SEARch:TRIGger:MPHY:ERRor:LENGth` on page 2315

13.24.5.2 M-PHY search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote commands:

- `SEARch:RESult:MPHY:FCOunt?` on page 2315
- `SEARch:RESult:MPHY:FRAMe<m>:CCOunt?` on page 2315

- SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:DATA? on page 2316
- SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:TYPE? on page 2316
- SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:STARt? on page 2316
- SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:STOP? on page 2317
- SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:STATe? on page 2317
- SEARch:RESult:MPHY:FRAMe<m>:DATA? on page 2317
- SEARch:RESult:MPHY:FRAMe<m>:FTYPe? on page 2318
- SEARch:RESult:MPHY:FRAMe<m>:STARt? on page 2319
- SEARch:RESult:MPHY:FRAMe<m>:STATe? on page 2319
- SEARch:RESult:MPHY:FRAMe<m>:STOP? on page 2320

13.24.5.3 Searching M-PHY

Prerequisite: A serial bus is configured for the M-PHY signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Tap "Setup search".
3. If the dialog box does not contain a search entry, tap the "Add" icon to create one, as described in "To create a user-defined search" on page 434.
4. Specify the search criteria according to Chapter 13.24.5.1, "M-PHY search setup", on page 912.
5. To acquire a waveform, press [Single].

   The R&S RTO6 performs an M-PHY decode according to the thresholds and protocol settings of the associated serial bus source.
6. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

   The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.24.5.4 SSIC search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"

Search criteria
Use the "Search criteria" dialog to define the search type. If you search for frame content, define the frame type in which data is to be searched.

You can define individual search parameters for the fields of the selected frame in the table below the "Search criteria". To specify these parameters, select a field in the table and define the data and/or index operators and values, or the bit state.

For a description of how to set the search conditions, see Chapter 13.24.3.3, "SSIC trigger settings", on page 908.

Remote command:
`SEARch:TRIGger:MPHY:SSIC:ERRor<m>:ENABle` on page 2325
`SEARch:TRIGger:MPHY:SSIC:FIENable` on page 2326
`SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:ENABle` on page 2326
`SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:BIT` on page 2327
`SEARch:TRIGger:MPHY:SSIC:BIT` on page 2327
`SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DMAX` on page 2328
`SEARch:TRIGger:MPHY:SSIC:DMAX` on page 2328
`SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DMIN` on page 2328
`SEARch:TRIGger:MPHY:SSIC:DMIN` on page 2328
`SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DOPerator` on page 2329
`SEARch:TRIGger:MPHY:SSIC:DOPerator` on page 2329
`SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:ENABle` on page 2326
`SEARch:TRIGger:MPHY:SSIC:FRENable` on page 2326
`SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IMAX` on page 2330
`SEARch:TRIGger:MPHY:SSIC:IMAX` on page 2330
`SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IMIN` on page 2330
`SEARch:TRIGger:MPHY:SSIC:IMIN` on page 2330
`SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IOPerator` on page 2330
`SEARch:TRIGger:MPHY:SSIC:IOPerator` on page 2330

13.25 8b/10b (option R&S RTO6-K590)

The 8b/10b coding scheme is used for high-speed serial data transmission, used by different standards, like fibre channel, PCIe, DVI and USB 3.0. Also the MIPI UniPro M-PHY interface in smartphones and display interfaces such as HDMI use 8b/10b encoding, as do base stations based on the Common Public Radio Interface (CPRI).

The R&S RTO6 can decode differential and single-ended 8b/10b signals. Furthermore, you can search the decoded data to isolate data contents and error states such as disparity violations.

You can also trigger on defined data contents and on error states.

- The 8b/10b code 922
- 8b/10b configuration 922
- 8b/10b trigger 927
- 8b/10b decode results 930
- Search on decoded 8b/10b data 931

13.25.1 The 8b/10b code

The 8b10b is a code where an 8-bit parallel data input is mapped into a 10-bit output. This code achieves a DC-balance and bounded disparity and simultaneously allows a reasonable clock recovery.

In the coding scheme, the original 8 bits are split into two blocks, three most significant bits and five least significant bits, see Figure 13-49. After that the 3-bit block is encoded into 4 bits and the 5-bit block into 6 bits. The 4-bit and the 6-bit blocks are then combined into a 10-bit.

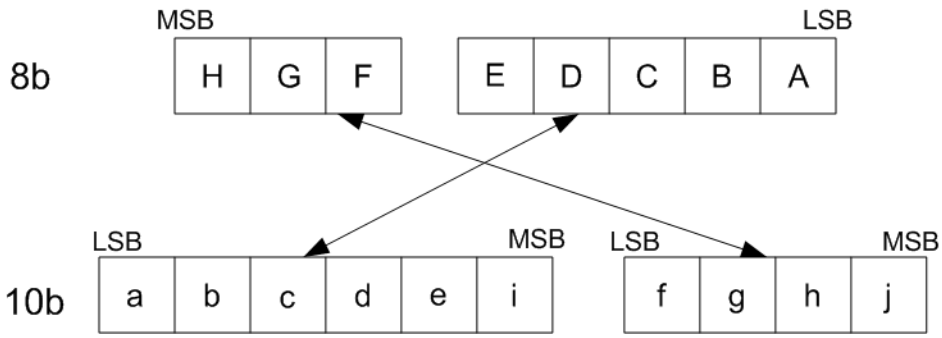


Figure 13-49: Coding scheme of the 8b/10b

The additional two bits I and j that join the stream are variable and ensure the running disparity.

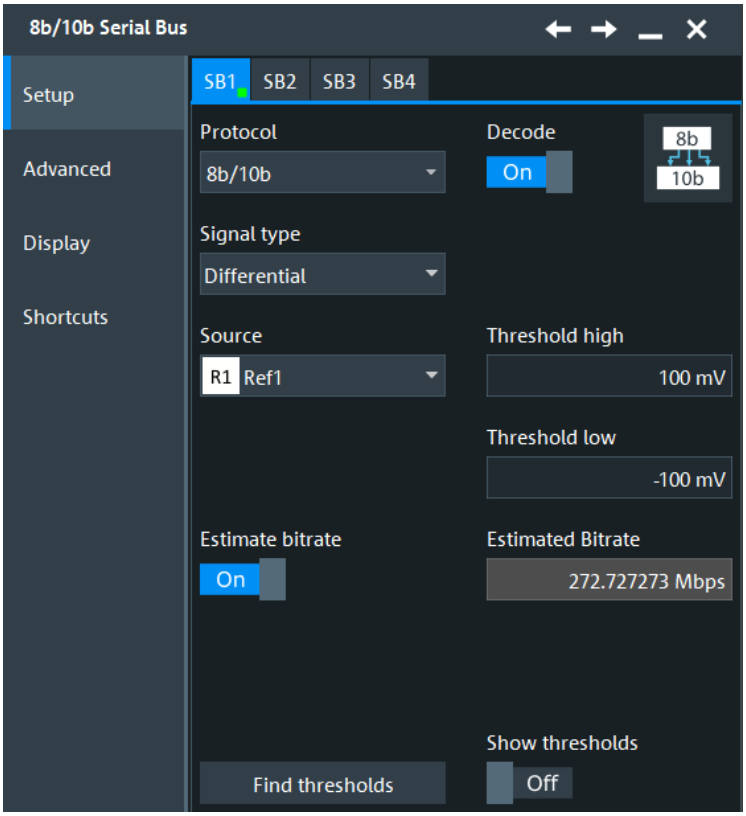
There are two types of characters used in the 8b/10b coding, special characters Kx.y and data characters Dx.y. x denotes the decimal value of EDCBA and is within the range from 0 to 31. y denotes the decimal value of HGF and is within the range 0 to 7.

13.25.2 8b/10b configuration

13.25.2.1 8b/10b configuration settings

Access: [Bus] > "8b/10b" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Signal type
Selects the signal type that is used for the decoding. You can choose between a differential and single-ended signal.

Remote command:
`BUS<m>:EBTB:TYPE` on page 2335

Source
Selects the source of the provided differential signal , if "Signal type" > "Differential".

Remote command:
`BUS<m>:EBTB:DIFFerential:SOURce` on page 2332

Source D+
Selects the source of the provided single ended signal, if "Signal type" > "Single ended".

Remote command:
`BUS<m>:EBTB:DPLus:SOURce` on page 2334

Source D-
Selects the source of the provided single ended signal , if "Signal type" > "Single ended".

Remote command:
`BUS<m>:EBTB:DMINus:SOURce` on page 2333

Threshold High
Sets the threshold high of the signal, if "Signal type" > "Differential".

Remote command:
`BUS<m>:EBTB:DIFFerential:THRHigh` on page 2332

Threshold Low
Sets the threshold low of the signal, if "Signal type" > "Differential".

Remote command:
`BUS<m>:EBTB:DIFFerential:THRLow` on page 2333

Threshold D+
Sets the high threshold (D+) of the signal, if "Signal type" > "Single ended".

Remote command:
`BUS<m>:EBTB:DPLus:THReshold` on page 2334

Threshold D-
Sets the high threshold (D-) of the signal, if "Signal type" > "Single ended".

Remote command:
`BUS<m>:EBTB:DMINus:THReshold` on page 2333

Find thresholds
Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
`BUS<m>:SETReflevels` on page 1671

Bitrate
Sets the number of transmitted bits per second.

Remote command:
`BUS<m>:EBTB:BITRate` on page 2332

Estimate Bitrate
Starts a software algorithm for the automatic determination of the bitrate. The value is displayed in the "Estimated Bitrate".

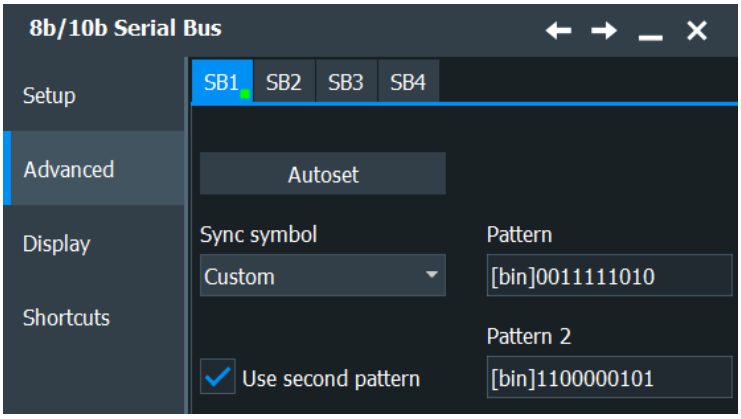
Remote command:
`BUS<m>:EBTB:BITDetermi` on page 2336
`BUS<m>:EBTB:EBTRate?` on page 2332

Show thresholds
If enabled, the threshold levels are displayed in the diagram.

Remote command:
`BUS<m>:THReshold` on page 1671

13.25.2.2 Advanced settings

Access: [Bus] > "Advanced" tab.



Autoset
Starts software algorithms for determining the signal threshold levels and bitrate.

Remote command:
`BUS<m>:FAUToset` on page 1671

Sync symbol
Selects the sync symbol, a control symbol used for low-level control functions. You can select one of the comma control symbols (K28.1, K28.5, K28.7) used for synchronization or you can enter a pattern.

Remote command:
`BUS<m>:EBTB:SYNC` on page 2335

Pattern
Selects a pattern that serves as a sync symbol.

Remote command:
`BUS<m>:EBTB:FCSY` on page 2334

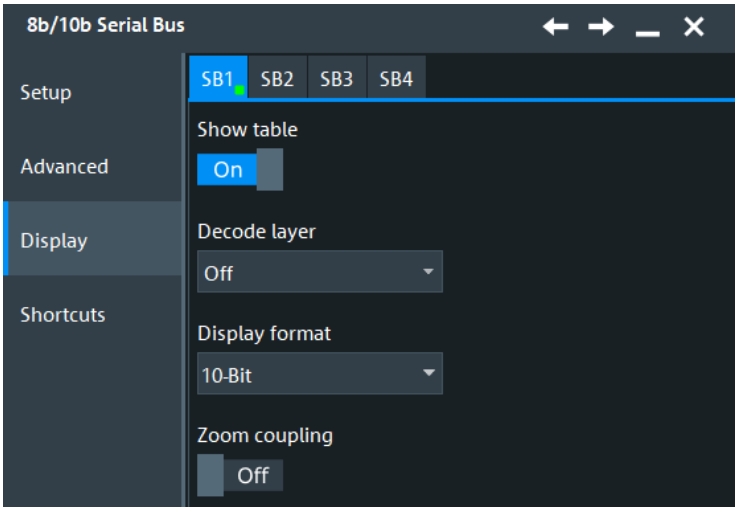
Pattern 2
Enables a second pattern that serves as a sync symbol and sets its value.

Remote command:
`BUS<m>:EBTB:SCSY` on page 2335
`BUS<m>:EBTB:USCS` on page 2335

13.25.2.3 Display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.



Display Format
Selects the display format for the results of the display table.

Remote command:
`BUS<m>:EBTB:DISF` on page 2333

13.25.2.4 Configuring 8b/10b

For configuration, you assign the line to the input channel, set the threshold, the bitrate, and the sync symbol.

For details on configuration settings, see Chapter 13.25.2.1, "8b/10b configuration settings", on page 922.

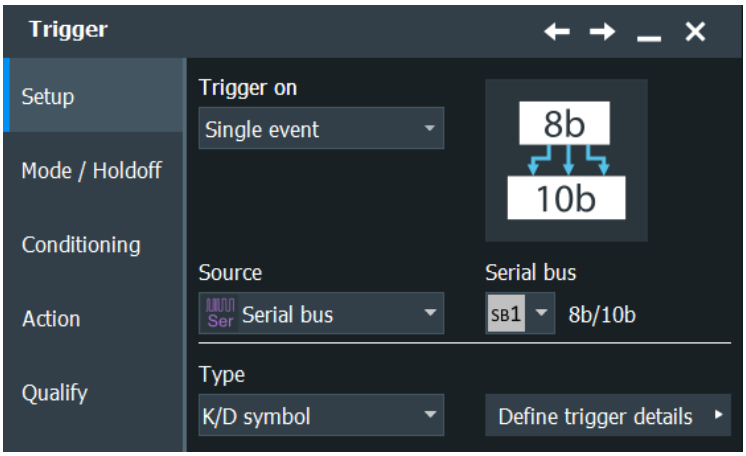
1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. In the "Setup" tab, set "Protocol"="8b/10b".
4. Tap the "Signal type" button, and select the used signal type for the measurement.
5. Select the source for the signal type.
6. Tap "Autoset".
7. Check the threshold and bitrate settings. Adjust the values if necessary.
8. Select the "Sync symbol" to be used for decoding.
9. Enable "Decode".

13.25.3 8b/10b trigger

If you need information on how to get started with triggering on 8b/10b signals, see Chapter 13.25.3.2, "Triggering on 8b/10b data", on page 930. Otherwise proceed with the 8b/10b trigger settings.

13.25.3.1 8b/10b trigger setup

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



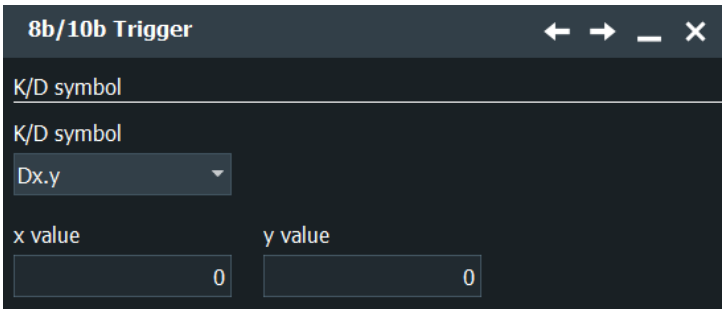
Type 8b/10b
Selects the type of condition to be triggered on. Available are:

- "K/D symbol"
- "8 bit pattern"
- "10 bit pattern"
- "Expression"
- "Error Condition"

Press "Define trigger details" to open a dialog for defining trigger type dependent settings.

Remote command:
`TRIGger<m>:EBTB:STYPe` on page 2338

K/D symbol
Selects the data character (Dx.y) or control character to be triggered on.



K/D Codes Symbol ← K/D symbol
Selects the control character.

Remote command:
`TRIGger<m>:EBTB:SYMType` on page 2338

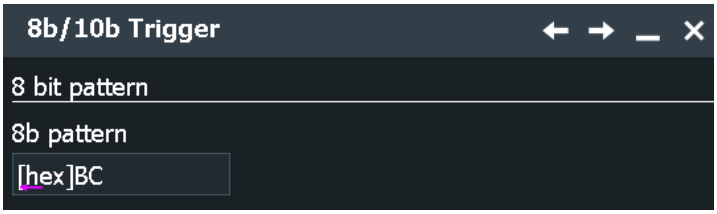
x value ← K/D symbol
Sets the x value of the data character Dx.y to be triggered on, if "K/D symbol" > "Dx.y".

Remote command:
`TRIGger<m>:EBTB:DX` on page 2337

y value ← K/D symbol
Sets the y value of the data character Dx.y to be triggered on, if "K/D symbol" > "Dx.y".

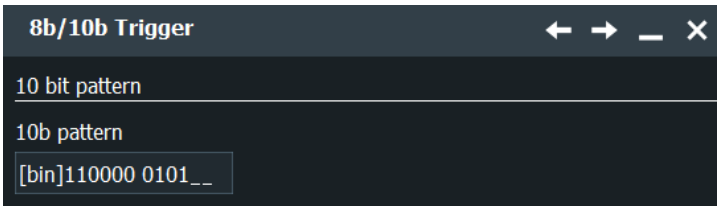
Remote command:
`TRIGger<m>:EBTB:DY` on page 2337

8 bit pattern
Sets the 8-bit pattern to be triggered on, if "Trigger Type" > "8 bit pattern".



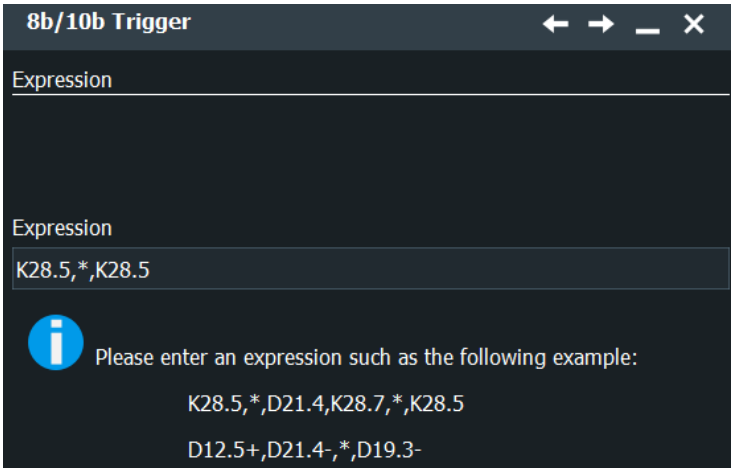
Remote command:
`TRIGger<m>:EBTB:EBPA` on page 2337

10 bit pattern
Sets the 10-bit pattern to be triggered on, if "Trigger Type" > "10 bit pattern".



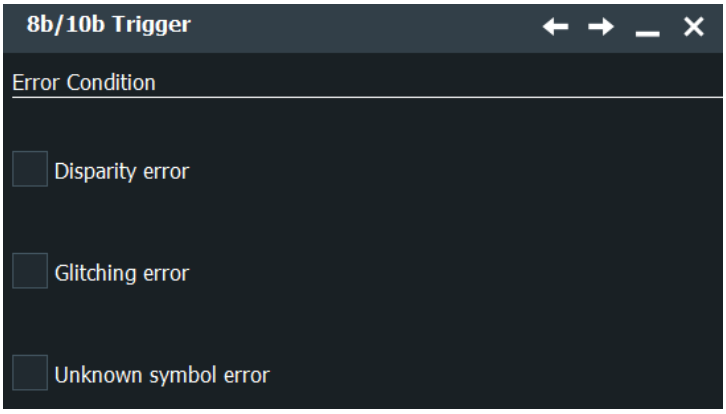
Remote command:
`TRIGger<m>:EBTB:TBPA` on page 2338

Expression
Defines the expression to be triggered on, if "Trigger Type" > "Expression".



Remote command:
`TRIGger<m>:EBTB:SYME` on page 2338

Error Condition
Triggers on the selected error conditions.



Disparity error ← Error Condition
Triggers on disparity errors.

Remote command:
`TRIGger<m>:EBTB:DISParityerr` on page 2336

Glitching error ← Error Condition
Triggers on glitch errors.

Remote command:
`TRIGger<m>:EBTB:GLITcherror` on page 2337

Unknown symbol error ← Error Condition
Triggers on a symbol that is not defined by the 8b/10b.

Remote command:
`TRIGger<m>:EBTB:UNK` on page 2338

13.25.3.2 Triggering on 8b/10b data

Prerequisite: A bus is configured for the 8b/10b signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Tap "Serial bus" and select the serial bus that is set to 8b/10b.
5. Tap "Type" and select the trigger type to be used for 8b/10b protocol analysis.
6. Depending on the selected trigger type, more setup conditions have to be specified.

For information on how to proceed with the configuration settings, see Chapter 13.25.3.1, "8b/10b trigger setup", on page 927.

13.25.4 8b/10b decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

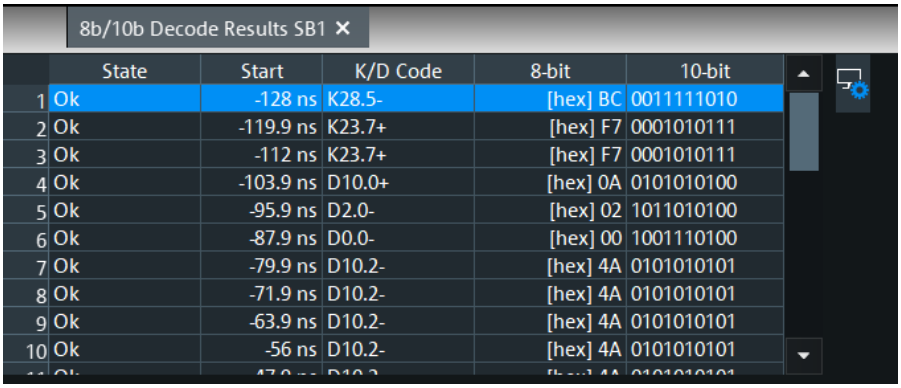


Table 13-48: Content of the decode result table

Column	Description
Symbol	Shows the index number of the symbol
State	Overall state of the symbol
Start	Time of the symbol start in relation to the trigger point
K/D Code	The K/D code
8-bit	The 8-bit value
10-bit	10-bit output value

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

Remote commands

Remote commands are described in Chapter 24.17.26.3, "Decode results", on page 2339.

13.25.5 Search on decoded 8b/10b data

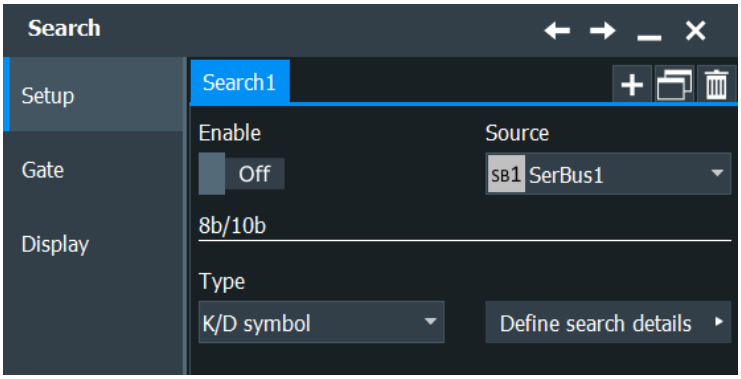
Using the search functionality, you can find various events in the decoded data, the same events which you also can trigger on. Before you can start the search, you have to configure the bus correctly and acquire decoded data.

To search on decoded data, set the search "Source" to the serial bus that is configured for the protocol to be analyzed.

See also Chapter 11, "Search functions", on page 419.

13.25.5.1 8b/10b search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"



Type
Selects the type of condition to be searched for. Available are:

- "K/D symbol"
- "8 bit pattern"
- "10 bit pattern"
- "Expression"
- "Error Condition"

Press "Define search details" to open a dialog for defining search type dependent settings.

Remote command:
`TRIGger<m>:EBTB:STYPe` on page 2338

K/D symbol
Selects the data character (Dx.y) or control character to be searched for.

K/D Codes Symbol ← K/D symbol
Selects the data character (Dx.y) or control character to bea searched for. You can specify the value of the data character to be searched for with "Dx Value" and "Dy Value".

Remote command:
`SEARch:TRIGger:EBTB:SYMType` on page 2343

y value ← K/D symbol
Sets the y value of the data character Dx.y to be searched for, if "K/D symbol" > "Dx.y".

Remote command:
`SEARch:TRIGger:EBTB:DY` on page 2342

x value ← K/D symbol
Sets the x value of the data character Dx.y to be searched for, if "K/D symbol" > "Dx.y".

Remote command:
`SEARch:TRIGger:EBTB:DX` on page 2342

Error Condition
Searches for the selected error conditions.

Disparity error
Searches for disparity errors.

Remote command:
`SEARch:TRIGger:EBTB:DISParityerr` on page 2341

Glitching error
Searches for glitching errors.

Remote command:
`SEARch:TRIGger:EBTB:GLITcherror` on page 2342

Unknown symbol error
Searches for a symbol that is not defined by the the 8b/10b.

Remote command:
`SEARch:TRIGger:EBTB:UNK` on page 2344

8b Pattern
Sets the 8-bit pattern to be searched for, if "Type" > "8-bit".

Remote command:
`SEARch:TRIGger:EBTB:EBPA` on page 2342

10b Pattern
Sets the 10-bit pattern to be searched for, if "Type" > "10-bit".

Remote command:
`SEARch:TRIGger:EBTB:TBPA` on page 2344

Expression
Defines the expression to be searched for, if "Type" > "Expression".

Remote command:
`SEARch:TRIGger:EBTB:SYME` on page 2343

13.25.5.2 8b/10b search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

Remote Commands:

- `SEARch:RESult:EBTB:SCOunt?` on page 2345
- `SEARch:RESult:EBTB:SYMBol<m>:DATA?` on page 2345
- `SEARch:RESult:EBTB:SYMBol<m>:STARt?` on page 2345
- `SEARch:RESult:EBTB:SYMBol<m>:STATus?` on page 2345

- `SEARch:RESult:EBTB:SYMBol<m>:STOP?` on page 2346

13.25.5.3 Searching 8b/10b data

Prerequisite: A serial bus is configured for the 8b/10b signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Tap "Setup search".
3. If the dialog box does not contain a search entry, tap the "Add" icon to create one, as described in "To create a user-defined search" on page 434.
4. Tap "Source" and select the serial bus that is set to 8b/10b (e.g. "SB1", unless already selected).

   The search dialog for 8b/10b protocol analysis is opened.
   There are no additional search criteria to be specified.
5. Tap "Source" and select the serial bus that is set to 8b/10b (e.g. "SB1", unless already selected).
6. Specify search criteria according to Chapter 13.25.5.1, "8b/10b search setup", on page 931.
7. To acquire a waveform, press [Single].

   The R&S RTO6 performs a 8b/10b decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).
8. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

   The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.26 PCIe (option R&S RTO6-K72)

R&S RTO6-K590 is a firmware option that enables the R&S RTO6 to analyze Peripheral Component Interconnect Express (PCIe) signals encoded by the PCIe standard, generation 1 and 2.

For analysis, PCIe encoded signals can be triggered, decoded and searched.

- The PCIe protocol 935
- PCIe configuration 937
- PCIe digital signal processing (DSP) settings 940

- PCIe trigger....942
- PCIe Gen 1/2 decode results....952
- PCIe Gen 1/2 search....953

13.26.1 The PCIe protocol

The PCIe is a high-speed serial computer expansion bus standard.

The communication between two PCIe devices is performed through logical connections called links. Each link consists of several lanes. The lanes contain one differential signaling pair for receiving data and the other for transmitting it.

PCIe logical layers

The PCIe has three logical layers:

- Transaction layer: assembles and disassembles transaction layer packets (TLPs). TLP transfer information like read and write and some event types.
- Data link layer: used for link management and data integrity. Error detection and correction are also done in this layer. The data link layer produces data link layer packets (DLLP).
- Physical layer: includes the circuitry for the interface operation.

PCIe Packet Types

The communication between the layers is done through packets. All packet types that can be decoded by the R&S RTO6 are listed below.

SCPI name	Description	Type
MRD32 / MRD64	Memory Read Request for 32 bit/ 64-bit addresssing packet format	TLP
MRDLK32 / MRDLK64	Memory Read Request-Locked for 32 bit/ 64-bit addresssing packet format	TLP
MWR32 / MWR64	Memory Write Request for 32 bit/ 64-bit addresssing packet format	TLP
IORD	I/O Read Request	TLP
IOWR	I/O Write Request	TLP
CFGRD0 / CFGRD1	Configuration Read Type 0/1	TLP
CFGWR0 / CFGWR1	Configuration Write Type 0/1	TLP
TCfgRd / TCfgWr	Deprecated TLP Type	TLP
MSG	Message Request	TLP
MSGD	Message Request with data payload	TLP
CPL	Completion without Data	TLP
CPLD	Completion with Data	TLP

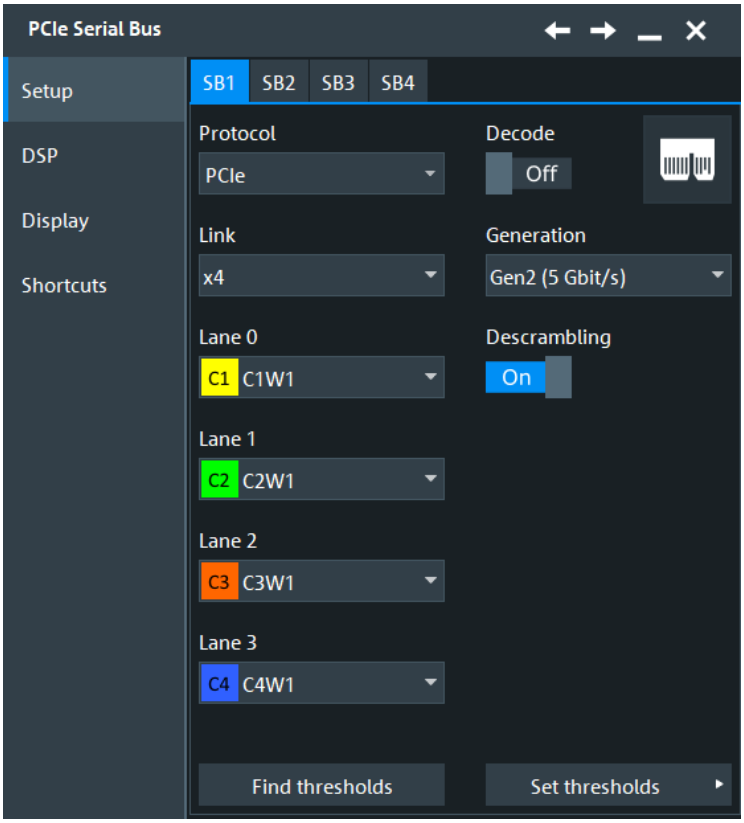
SCPI name	Description	Type
CPLLK	Completion for Locked Memory Read without Data	TLP
CPLDLK	Completion for Locked Memory Read	TLP
FA32 / FA64	Fetch and Add AtomicOp Request for 32 bit/ 64-bit addresssing packet format	TLP
SWP32 / SWP64	Unconditional Swap AtomicOp Request for 32 bit/ 64-bit addresssing packet format	TLP
CAS32 / CAS64	Compare and Swap AtomicOp Request for 32 bit/ 64-bit addresssing packet format	TLP
LPRFX	Local TLP Prefix	TLP
EPRFX	End-End TLP Prefix	TLP
EDB		TLP
ACK	Acknowledgement	DLLP
NAK	Negative acknowledgment	DLLP
PMEL1 / PMEL23	Power management Enter L1/ L23	DLLP
PMASRL1	Power management Active State Request L1	DLLP
PMRA	Power management request acknowledgment	DLLP
VENDS	Vendor Specific	DLLP
IFC1P / IFC2P	Initialization flow control 1/2 posted requests	DLLP
IFC1NP / IFC2NP	Initialization flow control 1/2 non posted requests	DLLP
IFC1CPL / IFC2CPL	Initialization flow control 1/2 completions	DLLP
UPDFCP	Update flow control 1/2 posted requests	DLLP
UPDFCNP	Update flow control non posted requests	DLLP
UPDFCCPL	Update flow control completions	DLLP
SKPOS	SKP ordered set	Ordered Set
TS1OS / TS2OS	Training sequence 1/2 ordered set	Ordered Set
FTSOS	Fast training sequence ordered set	Ordered Set
EIOS	Electrical idle ordered set	Ordered Set
EIEOS	Electrical idle exit ordered set	Ordered Set
COMPL	Compliance pattern	Ordered Set
MCOMPL	Modified Compliance pattern	Ordered Set

13.26.2 PCIe configuration

13.26.2.1 PCIe configuration settings

Access: [Bus] > "PCIe" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Generation
Selects the generation of the PCIe technology.

"PCIe Gen1 (2.5 Gbit/s) "	Selects the PCIe generation 1.0, standard introduced in 2003. It has a transfer rate of 2.5 Gbit/s.
"PCIe Gen2 (5 Gbit/s) "	Selects the PCIe generation 2.0, standard introduced in 2007. It has a transfer rate of 5 Gbit/s.

Remote command:
`BUS<m>:PCIE:GEN` on page 2347

Link
Selects the link width, the number of lanes that are used for the transmission of the data.

Remote command:
`BUS<m>:PCIE:LNKW` on page 2348

Descrambling
Enables descrambling of the data.

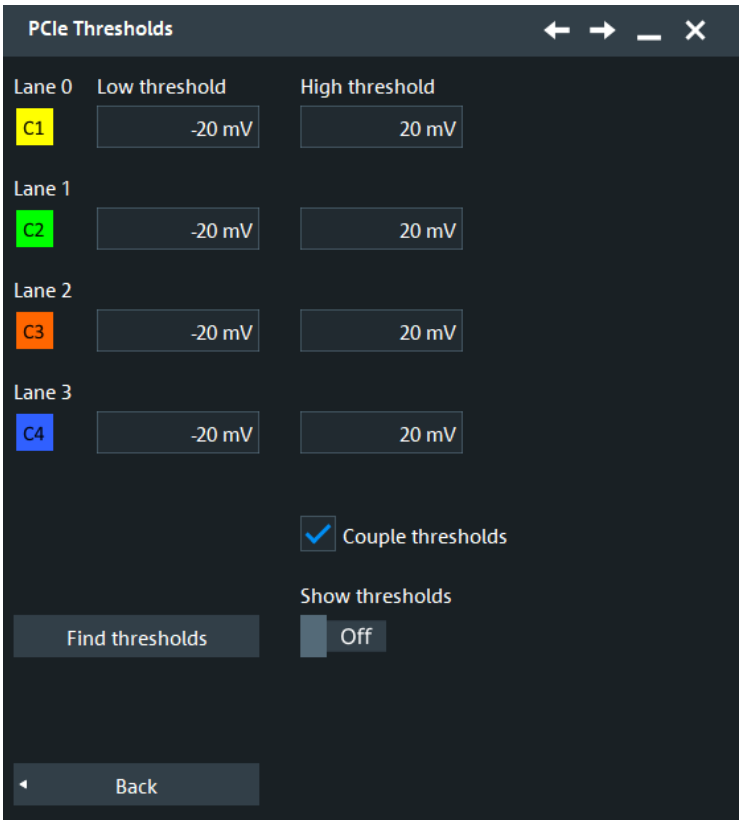
Remote command:
`BUS<m>:PCIE:DSCRambling` on page 2347

Lane 0/1/2/3
Select the signal sources for the logical lanes.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Remote command:
`BUS<m>:PCIE:LZER:SOURce` on page 2348
`BUS<m>:PCIE:LONE:SOURce` on page 2348
`BUS<m>:PCIE:LTWO:SOURce` on page 2348
`BUS<m>:PCIE:LTHRee:SOURce` on page 2348

Thresholds
Sets the threshold value for digitization of signals for each line.



Couple thresholds ← Thresholds
Enables the same threshold value for all lanes.

Find thresholds ← Thresholds
Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
`BUS<m>:SETReflevels` on page 1671

Low threshold ← Thresholds
Sets the lower threshold value for the respective lane.

Remote command:
`BUS<m>:PCIE:LONE:THRLow` on page 2349
`BUS<m>:PCIE:LTHRee:THRLow` on page 2349
`BUS<m>:PCIE:LTWO:THRLow` on page 2349
`BUS<m>:PCIE:LZER:THRLow` on page 2349

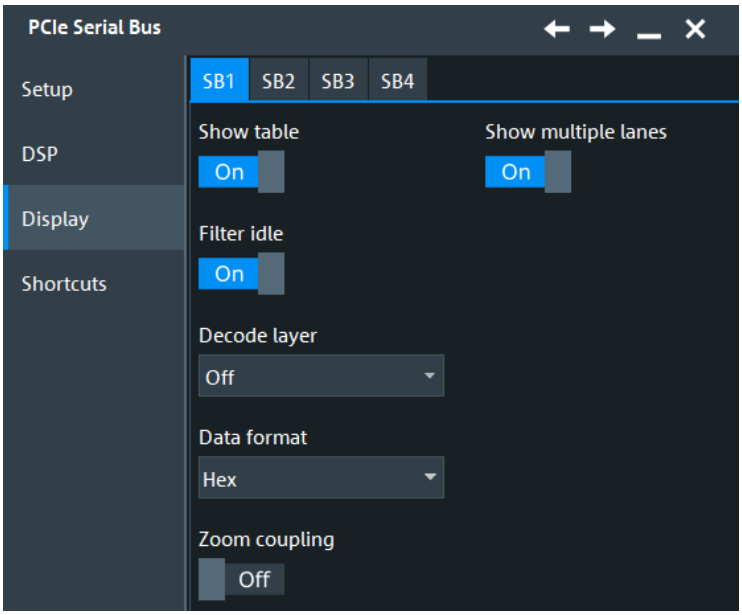
High threshold ← Thresholds
Sets the high threshold value for the respective lane.

Remote command:
`BUS<m>:PCIE:LONE:THRHigh` on page 2348
`BUS<m>:PCIE:LTHRee:THRHigh` on page 2348
`BUS<m>:PCIE:LTWO:THRHigh` on page 2348
`BUS<m>:PCIE:LZER:THRHigh` on page 2348

13.26.2.2 Display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.



Decode layer
Selects the decode layer.

Decoding is performed in several steps, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

Available are:
- "Off"
- "Edges"
- "Binary"

Filter Idle
Enables the filtering of idle packets.

Remote command:
`BUS<m>:PCIE:FIDLe` on page 2347

Show multiple lanes
Displays the result in multiple lanes.

Remote command:
`BUS<m>:PCIE:SMLanes` on page 2349

13.26.2.3 Configuring PCIe Gen 1/2

For configuration, you assign the line to the input channel, set the threshold, the bitrate, and the sync symbol.

For details on configuration settings, see Chapter 13.26.2.1, "PCIe configuration settings", on page 937.

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up.
3. In the "Setup" tab, select the protocol: "PCIe".
4. Tap the "Generation" button and select the PCIe protocol.
5. Tap the "Link" button and chose the link width.
6. Select the source for of the signal for each lane.
7. Check the threshold settings for each lane. Adjust the values if necessary.

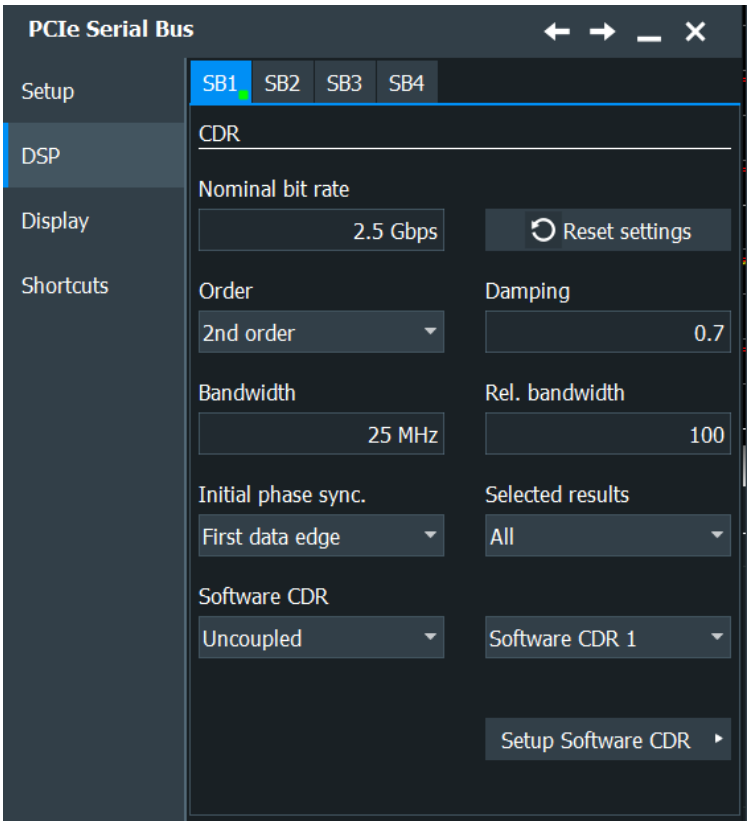
13.26.3 PCIe digital signal processing (DSP) settings

13.26.3.1 PCIe Gen 1/2 CDR configuration settings

The process of clock data recovery (CDR) generates a reference clock from a high-speed serial data stream that is sent without a dedicated clock signal. The generated

clock signal matches the frequency and is aligned to the phase of the data stream. The recovered clock can be used to sample the data stream and to obtain the sequence of transmitted bits.

Access: [Bus] > "Setup" tab > "Protocol = PCIe" = "Generation = Gen1/Gen2" > "DSP" tab



Nominal bit rate
Sets the quiescent frequency of the PLL. It corresponds to the data rate of the data stream from which the clock is to be recovered.

Remote command:
`BUS<m>:CDR:BITRate` on page 2349

PLL settings
Phase-locked loop parameters are listed below.

Note: Nomial bit rate, bandwidth and relative bandwidth are interacting settings. Modifying one parameter also changes one of the dependent parameters.

"Order"	Sets the order of the PLL: first or second order. PLL of higher order can compensate for more complex jitter behavior.
"Bandwidth"	Sets the PLL bandwidth. It defines the part of the spectrum that the PLL can follow during synchronization. The PLL bandwidth is usually defined by the transmission standard.
"Rel. bandwidth"	Sets the relative bandwidth, that is the ratio of the nominal bit rate to the PLL bandwidth.

"Damping" Sets the damping factor, which is only relevant for second order PLL.

Remote command:
`BUS<m>:CDR:PLL:ORDer` on page 2350
`BUS<m>:CDR:PLL:BWIDth` on page 2349
`BUS<m>:CDR:PLL:RELBwidth` on page 2350
`BUS<m>:CDR:PLL:DAMPing` on page 2350

Initial phase sync.
Defines the phase reference for the first clock edge.

"First sample" The first clock edge matches the first sample of the waveform at the left border of the display.

"First data edge" The first clock edge matches the first edge of the data signal.

Remote command:
`BUS<m>:CDR:SYNC` on page 2351

Selected results
The PLL requires some time to synchronize to the phase of the data stream. You can select when the CDR algorithm returns clock edges:

"After initial sync." The clock edges of the synchronization time are discarded; results are gathered after initial synchronization of the CDR. Thus, meaningful TIE measurement results can be obtained.

"All" All clock edges are used.

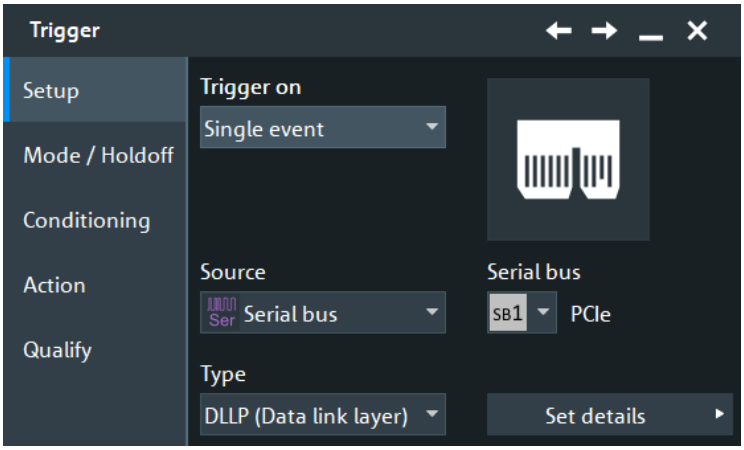
Remote command:
`BUS<m>:CDR:RESults` on page 2350

Software CDRs
Couples the protocol CDR to the selected software CDR. For more information about it see, Chapter 18.3, "Clock data recovery", on page 1112.

13.26.4 PCIe trigger

13.26.4.1 PCIe trigger settings

Access: [Bus] > "Shortcuts" tab > "Setup trigger"



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Trigger type PCI express

Selects the trigger type for the PCI express analysis.

"TLP (Transaction layer)"	Sets the trigger on a defined type of the transaction layer packet.
"DLLP (Data link layer)"	Sets the trigger on a defined type of the data link layer packet.
"Ordered Sets"	Sets the trigger on a defined ordered set.
"Error Condition"	Sets the trigger on a selection of error conditions.

Remote command:

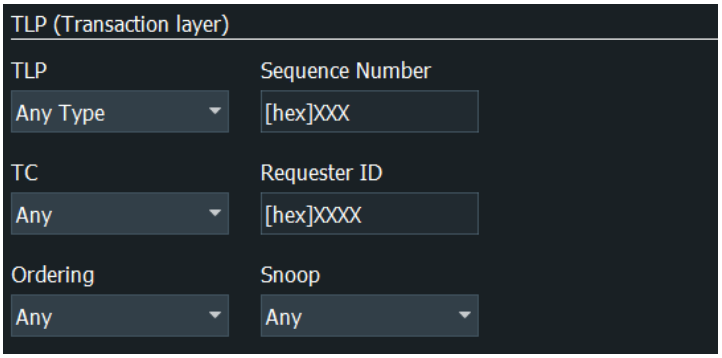
`TRIGger<m>:PCIE:TYPE` on page 2359

TLP (Transaction layer)

Specifies the trigger conditions for the transaction layer type trigger.

TLP ← TLP (Transaction layer)

Triggers on a transaction type.



"Any type"	Triggers on any transaction type.
"Mem"	Triggers on memory transaction type used for data transfers to memory-mapped locations.
"IO"	Triggers on I/O transaction type used for data transfers to I/O-mapped locations.
"Cfg"	Triggers on configuration transaction type used to set up the device functions.
"Msg"	Triggers on message transaction type used for transmitting diverse messages.
"Cpl"	Triggers on completion without data.
"FetchAdd"	Triggers on fetch and add atomic op request.
"Swap"	Triggers on unconditional swap atomic op request.
"CAS"	Triggers on compare and swap atomic op request transaction type.

Remote command:
`TRIGger<m>:PCIE:TLP:TYPE` on page 2359

Sequence Number ← TLP (Transaction layer)
Triggers on a sequence number, which indicates if any TLPs have been lost.

Remote command:
`TRIGger<m>:PCIE:TLP:SNUM` on page 2359

TC ← TLP (Transaction layer)
Triggers on a traffic class which is mapped on a specific virtual channel. The TC label is transmitted unmodified through the fabric.

Remote command:
`TRIGger<m>:PCIE:TLP:TCHN` on page 2359

Ordering ← TLP (Transaction layer)
Triggers on an ordering. The ordering can be relaxed or strong.

Remote command:
`TRIGger<m>:PCIE:TLP:ORDE` on page 2358

Snoop ← TLP (Transaction layer)
Triggers on a snoop state.

Remote command:
`TRIGger<m>:PCIE:TLP:SNOO` on page 2358

Requester ID ← TLP (Transaction layer)
Triggers on a requester ID, consisting of the requester's bus number, device number, and function number.

Remote command:
`TRIGger<m>:PCIE:TLP:REID` on page 2358

Device ID ← TLP (Transaction layer)
Triggers on a device ID, a field that can be assigned a value by the vendor for classifying root complex register blocks (RCRB). The Device ID is only valid for Configuration transaction. It is composed of 8-bit us number, 5-bit device number and a 3-bit function number.

Remote command:
`TRIGger<m>:PCIE:TLP:DEID` on page 2357

Read/Write ← TLP (Transaction layer)
Triggers on a read and/ or write state.

Remote command:
`TRIGger<m>:PCIE:TLP:MERW` on page 2358

Address Type ← TLP (Transaction layer)
Triggers on an address type. The address type can be 32 bit or 64 bit.

Remote command:
`TRIGger<m>:PCIE:TLP:ADRT` on page 2357

Type ← TLP (Transaction layer)
Triggers on a configuration type.

Remote command:
`TRIGger<m>:PCIE:TLP:CFGT` on page 2357

Routing ← TLP (Transaction layer)
Triggers on a selected message routing type.

Remote command:
`TRIGger<m>:PCIE:TLP:MSGR` on page 2358

Message Code ← TLP (Transaction layer)
Triggers on a specific message code.

Remote command:
`TRIGger<m>:PCIE:TLP:MSGC` on page 2358

Status ← TLP (Transaction layer)
Triggers on a completion status.

Remote command:
`TRIGger<m>:PCIE:TLP:CPLS` on page 2357

Completer ID ← TLP (Transaction layer)
Triggers on a completer ID, consisting of the completer's bus number, device number, and function number.

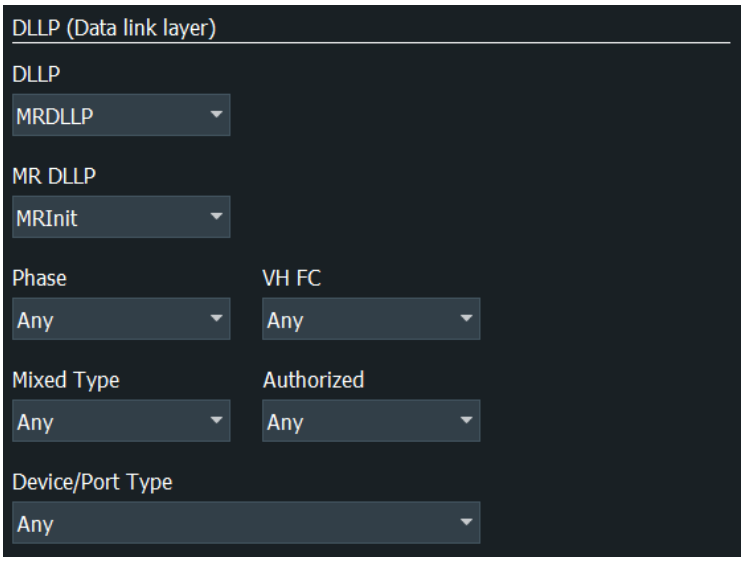
Remote command:
`TRIGger<m>:PCIE:TLP:CPID` on page 2357

DLLP Type
Sets the trigger to the type of DLLP. You can refine the settings for the different types according to their specifications.

Remote command:
`TRIGger<m>:PCIE:DLLP:TYPE` on page 2355

DLLP Any Type conditions
Sets the trigger to any type of DLLP.

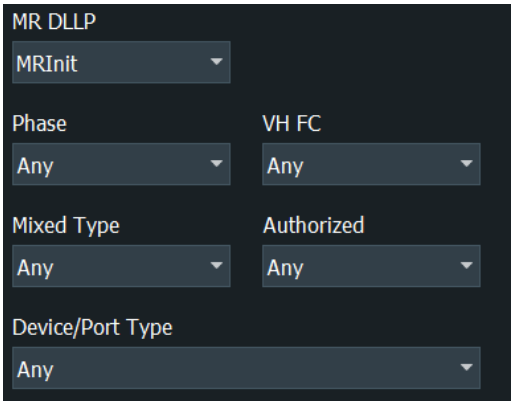
DLLP MRDLLP conditions
Specifies the conditions for the multi-root trigger.



MR DLLP ← DLLP MRDLLP conditions
Triggers on a multi-root type for the data link layer.

Remote command:
`TRIGger<m>:PCIE:DLLP:MULT:TYPE` on page 2354

MRInit ← DLLP MRDLLP conditions
Specifies the conditions for the multi-root initialization protocol trigger.



Phase ← MRInit ← DLLP MRDLLP conditions
Triggers on a phase state.

Remote command:
`TRIGger<m>:PCIE:DLLP:MULT:PHAS` on page 2353

VH FC ← MRInit ← DLLP MRDLLP conditions
Triggers on a virtual hierarchies flow control presence.

Remote command:
`TRIGger<m>:PCIE:DLLP:MULT:VHFC` on page 2354

Mixed Type ← MRInit ← DLLP MRDLLP conditions
Triggers on a mixed type presence.

Remote command:
`TRIGger<m>:PCIE:DLLP:MULT:MIXT` on page 2353

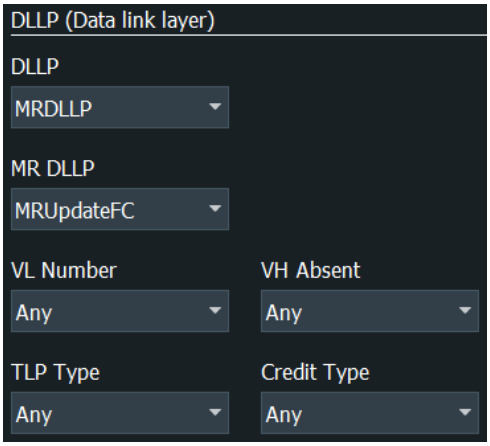
Authorized ← MRInit ← DLLP MRDLLP conditions
Triggers on an authorized state presence.

Remote command:
`TRIGger<m>:PCIE:DLLP:MULT:AUTH` on page 2352

Device/Port Type ← MRInit ← DLLP MRDLLP conditions
Triggers on an device/port type.

Remote command:
`TRIGger<m>:PCIE:DLLP:MULT:DVPT` on page 2353

MRInitFC1/InitFC2/UpdateFC ← DLLP MRDLLP conditions
Specifies the conditions for the multi-root initialization flow control protocol trigger.



VL Number ← MRInitFC1/InitFC2/UpdateFC ← DLLP MRDLLP conditions
Triggers on a virtual link (VL) number.

Remote command:
`TRIGger<m>:PCIE:DLLP:MULT:VLNR` on page 2355

VH Absent ← MRInitFC1/InitFC2/UpdateFC ← DLLP MRDLLP conditions
Triggers on absent virtual hierarchies (VH).

Remote command:
`TRIGger<m>:PCIE:DLLP:MULT:HABS` on page 2353

TLP Type ← MRInitFC1/InitFC2/UpdateFC ← DLLP MRDLLP conditions
Triggers on a transaction layer type for a multi-root data link layer trigger type.

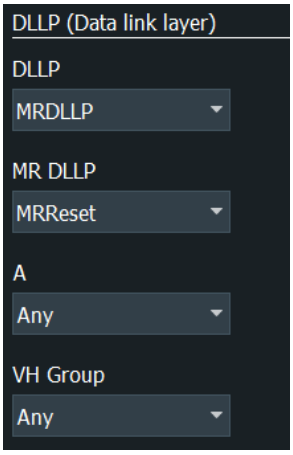
"Any"	Any transaction layer type.
"P"	Posted credit.
"N"	Non posted credit.
"Cpl"	Completion credit.

Remote command:
`TRIGger<m>:PCIE:DLLP:MULT:TLPT` on page 2354

Credit Type ← MRInitFC1/InitFC2/UpdateFC ← DLLP MRDLLP conditions
Triggers on any, data or header credit type.

Remote command:
`TRIGger<m>:PCIE:DLLP:MULT:CRET` on page 2353

MRReset ← DLLP MRDLLP conditions
Specifies the conditions for the multi-root reset protocol trigger.



A ← MRReset ← DLLP MRDLLP conditions
Triggers on the value of the A bit (Ack/Request).

Remote command:
`TRIGger<m>:PCIE:DLLP:MULT:RESA` on page 2353

VH Group ← MRReset ← DLLP MRDLLP conditions
Triggers on the selected virtual hierarchies group.

Remote command:
`TRIGger<m>:PCIE:DLLP:MULT:VHGR` on page 2354

DLLP ACK/NAK conditions
Triggers on an acknowledgement (ACK)/ negative acknowledgment (NAK).

Sequence ← DLLP ACK/NAK conditions
Triggers on the sequence field indicating what TLPs are affected by the ACK/NAK.

Remote command:
`TRIGger<m>:PCIE:DLLP:SEQ` on page 2355

DLLP Init FC1/InitFC2/UpdateFC conditions
Triggers on flow control initialization/update conditions.

Credit Type ← DLLP Init FC1/InitFC2/UpdateFC conditions
Triggers on a credit type value.

"Any"	Triggers on any credit type.
"P"	Triggers on posted requests.
"NP"	Triggers on non posted requests.
"CPL"	Triggers on completions.

Remote command:
`TRIGger<m>:PCIE:DLLP:FCTL:CRET` on page 2352

VC ID ← DLLP Init FC1/InitFC2/UpdateFC conditions
Triggers on a virtual channel ID value.

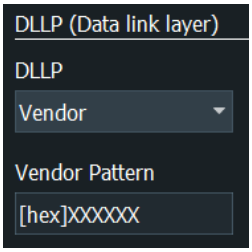
Remote command:
`TRIGger<m>:PCIE:DLLP:FCTL:VCID` on page 2352

DLLP PM conditions
Triggers on a specified power management conditions.

PM Type ← DLLP PM conditions
Triggers on a power management type.

Remote command:
`TRIGger<m>:PCIE:DLLP:POWM` on page 2355

DLLP Vendor conditions
Triggers on a specified vendor conditions.



Vendor Pattern ← DLLP Vendor conditions
Triggers on a vendor pattern.

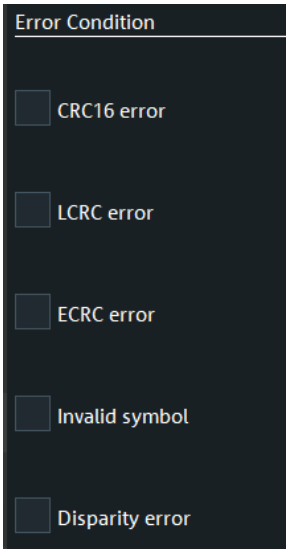
Remote command:
`TRIGger<m>:PCIE:DLLP:VPAT` on page 2355

Ordered set condition
Triggers on an ordered set.

"SKP OS"	Triggers on an SKP ordered sets.
"Training Seq 1/2"	Triggers on a training sequence ordered set.
"Fast Training Seq"	Triggers on a fast training sequence ordered set.
"Electrical Idle OS"	Triggers on an electrical idle ordered set that must be send by the transmitter before it can enter the electrical idle.
"Electrical Idle Exit OS"	Triggers on an electrical idle exit ordered set.
"Compliance Pattern"	Triggers on a compliance pattern ordered set.

Remote command:
`TRIGger<m>:PCIE:OSET:TYPE` on page 2356

Error Condition PCIe Gen 1/2
Triggers on enabled error condition.



"CRC16 error"	Triggers on 16-bit cyclic redundancy check (CRC) errors.
"LCRC error"	Triggers on link cyclic redundancy check (LCRC) errors.
"ECRC error"	Triggers on end-to-end cyclic redundancy checksum (ECRC) errors.
"Invalid symbol"	Triggers on invalid symbol errors.
"Disparity error"	Triggers on disparity errors.

Remote command:
`TRIGger<m>:PCIE:ERRC:CRC` on page 2355
`TRIGger<m>:PCIE:ERRC:DISP` on page 2356
`TRIGger<m>:PCIE:ERRC:ECRC` on page 2356
`TRIGger<m>:PCIE:ERRC:INVP` on page 2356
`TRIGger<m>:PCIE:ERRC:LCRC` on page 2356

13.26.4.2 Triggering on PCIe Gen 1/2

Prerequisite: A bus is configured for the PCIe Gen 1/2 signal to be analyzed.

For the basic trigger settings, proceed in the following way:

1. Press the [Bus] key and select the "Shortcuts" tab.
2. Press "Setup trigger".
3. Tap the "Source" button and select the "Serial bus" trigger source.
4. Tap "Serial bus" and select the serial bus that is set to PCIe Gen 1/2.
5. Tap "Trigger Type PCI express" and select the trigger type to be used for PCIe Gen 1/2 protocol analysis.
6. Depending on the selected trigger type, more setup conditions can be specified.

For information on how to proceed with the configuration settings, see Chapter 13.26.4.1, "PCIe trigger settings", on page 942.

13.26.5 PCIe Gen 1/2 decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Serial Bus" dialog > "Setup" tab, enable "Decode".
2. In the "Serial Bus" dialog > "Display" tab, select additional result display settings: "Show table" and "Show binary".
3. If required, enable "Zoom coupling"

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode results table

The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

Table 13-49: Content of the Decode results table

Column	Description
State	Overall state of the packet
Start	Start time of the packet
Type	Type of packet
Sequence count	Sequence count
Tag	Tag of the packet
Length	Length of packet
Address	Value of the address
HdrFC	Credit value of the header

Column	Description
DataFC	Credit value of the payload data
Completion status	Completion status field value
Requester ID	Requester ID
Completer ID	Completer ID
Bit rate	The bit rate of the packet

Table 13-50: Content of the Details frame table

Column	Description
State	State of the field
Type	Field type
Numeric	Hexadecimal value of the field
Formatted	Formatted content of the field

Tools

Press "Display settings" in the "Decode Results" table to access the "Display" tab.

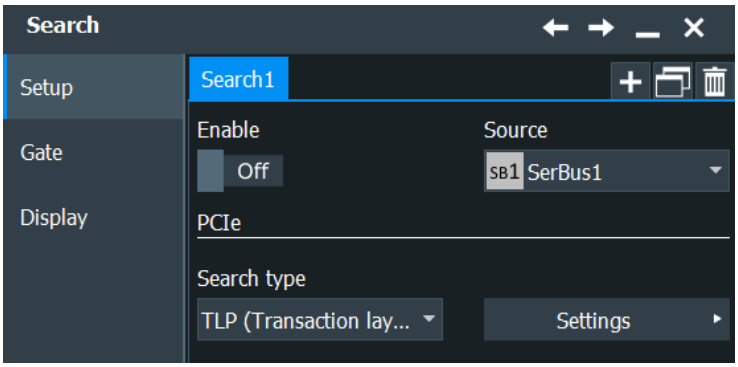
Remote commands

Remote commands to retrieve decode results are described in Chapter 24.17.27.4, "Decode results", on page 2359.

13.26.6 PCIe Gen 1/2 search

13.26.6.1 PCIe Gen 1/2 search settings

Access: [Bus] > "Shortcuts" tab > "Setup search"



Make sure that:

- In the [Bus] > "Setup" tab:
  - The correct "Protocol" is selected.
  - The data sources of the serial bus are channel signals.
- In the "Trigger" > "Setup" tab:
  - The "Trigger on" = "Single event"
  - The trigger "Source" = "Serial bus".
  - The correct "Serial bus" is selected.

Serial bus
Selects the serial bus to be triggered on. Make sure to select the correct bus before you enter the settings.

To trigger on a serial bus, the signals sources must be channel signals. If the data or clock source is a math or reference waveform, you cannot trigger on that bus.

Remote command:
`TRIGger<m>:SOURce:SBSelect` on page 1675

Protocol
Defines the protocol type of the selected serial bus.

Remote command:
`BUS<m>:TYPE` on page 1670

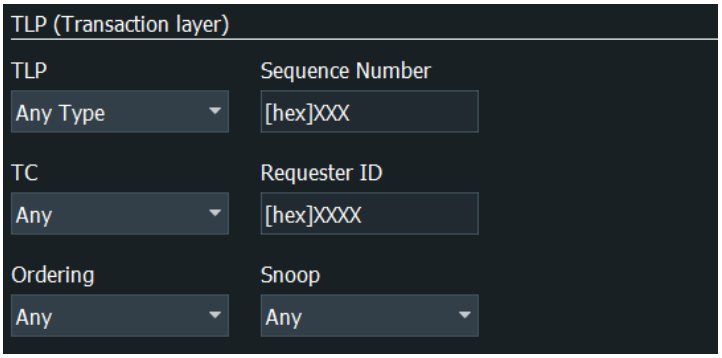
Searches for PCI express
Selects the search type for the PCI express analysis.

"TLP (Transaction layer)"	Sets the search on a defined type of the transaction layer packet.
"DLLP (Data link layer)"	Sets the search on a defined type of the data link layer packet.
"Ordered Sets"	Sets the search on a defined ordered set.
"Error Condition"	Sets the search on a selection of error conditions.

Remote command:
`SEARch:TRIGger:PCIE:TYPE` on page 2380

TLP (Transaction layer)
Specifies the search conditions for the transaction layer type search.

TLP ← TLP (Transaction layer)
Searches for a transaction type.



"Any type"	Searches for any transaction type.
"Mem"	Searches for memory transaction type used for data transfers to memory-mapped locations.
"IO"	Searches for I/O transaction type used for data transfers to I/O-mapped locations.
"Cfg"	Searches for configuration transaction type used to setup the device functions.
"Msg"	Searches for message transaction type used for transmitting diverse messages.
"Cpl"	Searches for completion without data.
"FetchAdd"	Searches for fetch and add atomic op request.
"Swap"	Searches for unconditional swap atomic op request.
"CAS"	Searches for compare and swap atomic op request transaction type.

Remote command:
`SEARch:TRIGger:PCIE:TLP:TYPE` on page 2380

Sequence Number ← TLP (Transaction layer)
Searches for a sequence number, which indicates if any TLPs have been lost.

Remote command:
`SEARch:TRIGger:PCIE:TLP:SNUM` on page 2379

TC ← TLP (Transaction layer)
Searches for a traffic class which is mapped on a specific virtual channel. The TC label is transmitted unmodified through the fabric.

Remote command:
`SEARch:TRIGger:PCIE:TLP:TCHN` on page 2379

Ordering ← TLP (Transaction layer)
Searches for an ordering. The ordering can be relaxed or strong.

Remote command:
`SEARch:TRIGger:PCIE:TLP:ORDE` on page 2378

Snoop ← TLP (Transaction layer)
Searches for a snoop state.

Remote command:
`SEARch:TRIGger:PCIE:TLP:SNOO` on page 2379

Requester ID ← TLP (Transaction layer)
Searches for a requester ID, consisting of the requester's bus number, device number, and function number.

Remote command:
`SEARch:TRIGger:PCIE:TLP:REID` on page 2379

Device ID ← TLP (Transaction layer)
Searches for a device ID, a field that can be assigned a value by the vendor for classifying root complex register blocks (RCRB). The Device ID is only valid for Configuration transaction. It is composed of 8-bit Bus number, 5-bit Device number and a 3-bit Function number.

Remote command:
`SEARch:TRIGger:PCIE:TLP:DEID` on page 2377

Read/Write ← TLP (Transaction layer)
Searches for a read and/ or write state.

Remote command:
`SEARch:TRIGger:PCIE:TLP:MERW` on page 2378

Address Type ← TLP (Transaction layer)
Searches for an address type. The address type can be 32 bit or 64 bit.

Remote command:
`SEARch:TRIGger:PCIE:TLP:ADRT` on page 2376

Type ← TLP (Transaction layer)
Searches for a configuration type.

Remote command:
`SEARch:TRIGger:PCIE:TLP:CFGT` on page 2377

Routing ← TLP (Transaction layer)
Searches for a selected message routing type.

Remote command:
`SEARch:TRIGger:PCIE:TLP:MSGR` on page 2378

Message Code ← TLP (Transaction layer)
Searches for a specific message code.

Remote command:
`SEARch:TRIGger:PCIE:TLP:MSGC` on page 2378

Status ← TLP (Transaction layer)
Searches for a completion status.

Remote command:
`SEARch:TRIGger:PCIE:TLP:CPLS` on page 2377

Completer ID ← TLP (Transaction layer)
Searches for a completer ID, consisting of the completer's bus number, device number, and function number.

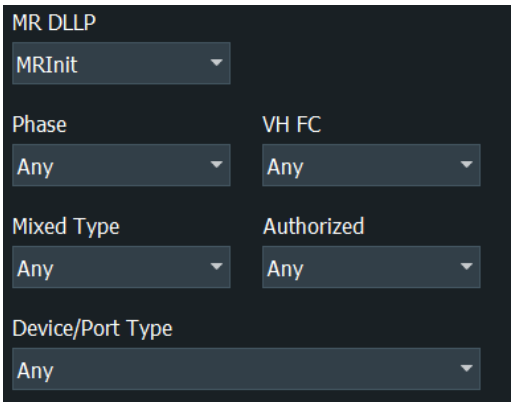
Remote command:
`SEARch:TRIGger:PCIE:TLP:CPID` on page 2377

DLLP Type
Sets the trigger to the type of DLLP. You can refine the settings for the different types according to their specifications.

Remote command:
`SEARch:TRIGger:PCIE:DLLP:TYPE` on page 2373

DLLP Any Type conditions
Sets the trigger to any type of DLLP.

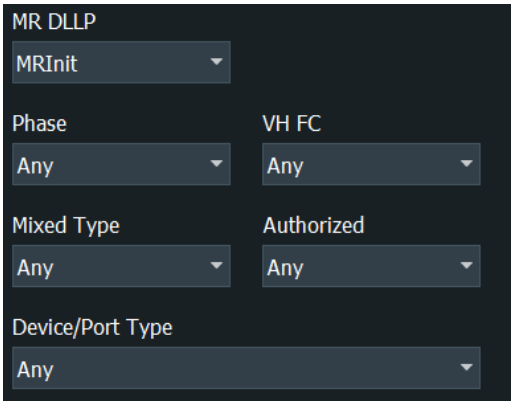
DLLP MRDLLP conditions
Specifies the conditions for the multi-root trigger.



MR DLLP ← DLLP MRDLLP conditions
Searches for a multi-root type for the data link layer.

Remote command:
`SEARch:TRIGger:PCIE:DLLP:MULT:TYPE` on page 2371

MRInit ← DLLP MRDLLP conditions
Specifies the conditions for the multi-root initialization protocol search.



Phase ← MRInit ← DLLP MRDLLP conditions
Searches for a phase state.

Remote command:
`SEARch:TRIGger:PCIE:DLLP:MULT:PHAS` on page 2370

VH FC ← MRInit ← DLLP MRDLLP conditions
Searches for a virtual hierarchies flow control presence.

Remote command:
`SEARch:TRIGger:PCIE:DLLP:MULT:VHFC` on page 2371

Mixed Type ← MRInit ← DLLP MRDLLP conditions
Searches for a mixed type presence.

Remote command:
`SEARch:TRIGger:PCIE:DLLP:MULT:MIXT` on page 2370

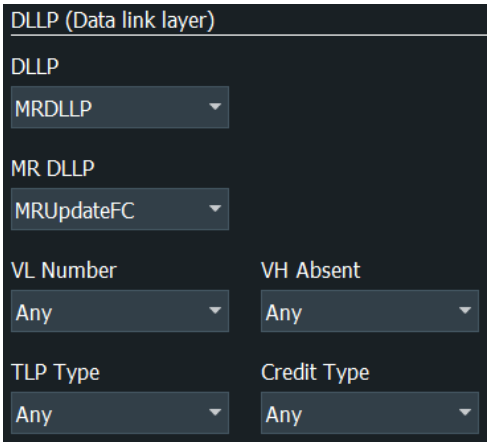
Authorized ← MRInit ← DLLP MRDLLP conditions
Searches for an authorized state presence.

Remote command:
`SEARch:TRIGger:PCIE:DLLP:MULT:AUTH` on page 2369

Device/Port Type ← MRInit ← DLLP MRDLLP conditions
Searches for an device/port type.

Remote command:
`SEARch:TRIGger:PCIE:DLLP:MULT:DVPT` on page 2370

MRInitInit FC1/InitFC2/UpdateFC ← DLLP MRDLLP conditions
Specifies the conditions for the multi-root initialization flow control protocol search.



VL Number ← MRInitInit FC1/InitFC2/UpdateFC ← DLLP MRDLLP conditions
Searches for a virtual link (VL) number.

Remote command:
`SEARch:TRIGger:PCIE:DLLP:MULT:VLNR` on page 2372

VH Absent ← MRInitInit FC1/InitFC2/UpdateFC ← DLLP MRDLLP conditions
Searches for absent virtual hierarchies (VH).

Remote command:
`SEARch:TRIGger:PCIE:DLLP:MULT:HABS` on page 2370

TLP Type ← MRInitInit FC1/InitFC2/UpdateFC ← DLLP MRDLLP conditions
Searches for a transaction layer type for a multi-root data link layer trigger type.

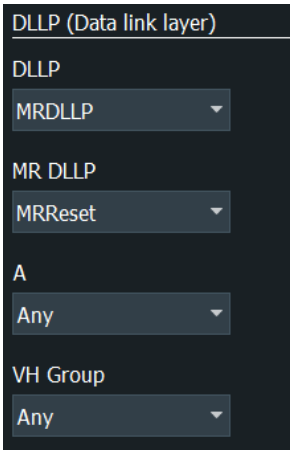
"Any"	Any transaction layer type.
"P"	Posted credit.
"N"	Non posted credit.
"Cpl"	Completion credit.

Remote command:
`SEARch:TRIGger:PCIE:DLLP:MULT:TLPT` on page 2371

Credit Type ← MRInitInit FC1/InitFC2/UpdateFC ← DLLP MRDLLP conditions
Searches for any, data or header credit type.

Remote command:
`SEARch:TRIGger:PCIE:DLLP:FCTL:CRET` on page 2369

MRReset ← DLLP MRDLLP conditions
Specifies the conditions for the multi-root reset protocol search.



A ← MRReset ← DLLP MRDLLP conditions
Searches for the value of the A bit (Ack/ Request).

Remote command:
`SEARch:TRIGger:PCIE:DLLP:MULT:RESA` on page 2371

VH Group ← MRReset ← DLLP MRDLLP conditions
Searches for the selected virtual hierarchies group.

Remote command:
`SEARch:TRIGger:PCIE:DLLP:MULT:VHGR` on page 2372

DLLP ACK/NAK conditions
Searches for an acknowledgement (ACK)/ negative acknowledgment (NAK).

Sequence ← DLLP ACK/NAK conditions
Searches for the sequence field indicating what TLPs are affected by the ACK/NAK.

Remote command:
`SEARch:TRIGger:PCIE:DLLP:SEQ` on page 2373

DLLP Init FC1/InitFC2/UpdateFC conditions
Searches for flow control initialization/update conditions.

Credit Type ← DLLP Init FC1/InitFC2/UpdateFC conditions
Searches for a credit type value.

"Any"	Searches for any credit type.
"P"	Searches for posted requests.
"NP"	Searches for non posted requests.
"CPL"	Searches for completions.

Remote command:
`SEARch:TRIGger:PCIE:DLLP:FCTL:CRET` on page 2369

VC ID ← DLLP Init FC1/InitFC2/UpdateFC conditions
Searches for a virtual channel ID value.

Remote command:
SEARch:TRIGger:PCIE:DLLP:FCTL:VCID on page 2369

DLLP PM conditions
Searches for a specified power management conditions.

PM Type ← DLLP PM conditions
Searches for a power management type.

Remote command:
SEARch:TRIGger:PCIE:DLLP:POWM on page 2372

DLLP Vendor conditions
Searches for a specified vendor conditions.

Vendor Pattern ← DLLP Vendor conditions
Searches for a vendor pattern.

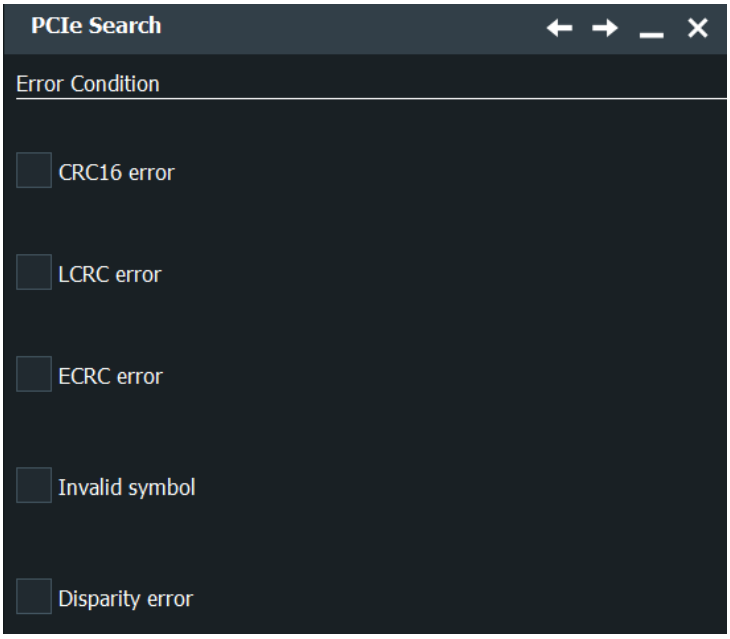
Remote command:
SEARch:TRIGger:PCIE:DLLP:VPAT on page 2373

Ordered set condition
Searches for the enabled ordered sets.

"SKP OS"	Searches for an SKP ordered sets.
"Training Seq 1/2"	Searches for a training sequence ordered set.
"Fast Training Seq"	Searches for a fast training sequence ordered set.
"Electrical Idle OS"	Searches for an electrical idle ordered set that must be send by the transmitter before it can enter the electrical idle.
"Electrical Idle Exit OS"	Searches for an electrical idle exit ordered set.
"Start of data sequence OS"	Available for PCIe "Generation" = " Gen 3 (8 Gbit/s)" Searches for a start of data sequence order set.
"Compliance Pattern"	Searches for a compliance pattern ordered set.

Remote command:
SEARch:TRIGger:PCIE:OSET:COMP on page 2375
SEARch:TRIGger:PCIE:OSET:EIDE on page 2375
SEARch:TRIGger:PCIE:OSET:EIDL on page 2375
SEARch:TRIGger:PCIE:OSET:FTS on page 2375
SEARch:TRIGger:PCIE:OSET:SKIP on page 2376
SEARch:TRIGger:PCIE:OSET:TSONe on page 2376
SEARch:TRIGger:PCIE:OSET:TSTWo on page 2376

Error Condition
Searches for enabled error conditions.



"CRC16 error"	Searches for 16-bit cyclic redundancy check (CRC) errors.
"LCRC error"	Searches for link cyclic redundancy check (LCRC) errors.
"ECRC error"	Searches for end-to-end cyclic redundancy checksum (ECRC) errors.
"Invalid symbol"	Searches for invalid symbol errors.
"Disparity error"	Searches for disparity errors.

Remote command:
`SEARch:TRIGger:PCIE:ERRC:CRC` on page 2373
`SEARch:TRIGger:PCIE:ERRC:DISP` on page 2374
`SEARch:TRIGger:PCIE:ERRC:ECRC` on page 2374
`SEARch:TRIGger:PCIE:ERRC:INVP` on page 2374
`SEARch:TRIGger:PCIE:ERRC:LCRC` on page 2374

13.26.6.2 PCIe Gen 1/2 search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

The results of the search are the same as for decoding, see Chapter 13.27.3, "DDR decode results", on page 971.

Remote Commands:

- SEARch:RESult:PCIE:PCOunt? on page 2386
- SEARch:RESult:PCIE:PACKet<m>:ADDR? on page 2381
- SEARch:RESult:PCIE:PACKet<m>:CPID? on page 2381
- SEARch:RESult:PCIE:PACKet<m>:CPS? on page 2381
- SEARch:RESult:PCIE:PACKet<m>:DATA? on page 2381
- SEARch:RESult:PCIE:PACKet<m>:DFC? on page 2382
- SEARch:RESult:PCIE:PACKet<m>:FCOunt? on page 2382
- SEARch:RESult:PCIE:PACKet<m>:HFC? on page 2383
- SEARch:RESult:PCIE:PACKet<m>:LEN? on page 2384
- SEARch:RESult:PCIE:PACKet<m>:RQID? on page 2384
- SEARch:RESult:PCIE:PACKet<m>:SEQN? on page 2384
- SEARch:RESult:PCIE:PACKet<m>:STARt? on page 2385
- SEARch:RESult:PCIE:PACKet<m>:STATe? on page 2385
- SEARch:RESult:PCIE:PACKet<m>:STOP? on page 2385
- SEARch:RESult:PCIE:PACKet<m>:TAG? on page 2385
- SEARch:RESult:PCIE:PACKet<m>:TYPE? on page 2386
- SEARch:RESult:PCIE:PACKet<m>:FCOunt? on page 2382
- SEARch:RESult:PCIE:PACKet<m>:FLD<n>:FVAL? on page 2382
- SEARch:RESult:PCIE:PACKet<m>:FLD<n>:STATus? on page 2382
- SEARch:RESult:PCIE:PACKet<m>:FLD<n>:TYPE? on page 2383
- SEARch:RESult:PCIE:PACKet<m>:FLD<n>:VAL? on page 2383

13.26.6.3 Searching for PCIe Gen 1/2

Prerequisite: A serial bus is configured for the PCIe signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press the [Bus] key and selet the "Shortcuts" tab.
2. Tap "Setup search".
3. If the dialog box does not contain a search entry, tap the "Add" icon to create one, as described in "To create a user-defined search" on page 434.
4. Tap "Source" and select the serial bus that is set to PCIe (e.g. "SB1").
5. Specify the search criteria according to Chapter 13.26.6.1, "PCIe Gen 1/2 search settings", on page 953.
6. To acquire a waveform, press [Single].

   The R&S RTO6 performs an PCIe decode according to the thresholds and protocol settings of the associated serial bus source.

7. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

   The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.27 DDR (option R&S RTO6-K91)

Double Data Rate Type 3 (DDR3) is a type of Synchronous Dynamic Random Access Memory (SDRAM). With option R&S RTO6-K91 you can perform the following analysis on DDR3 data:

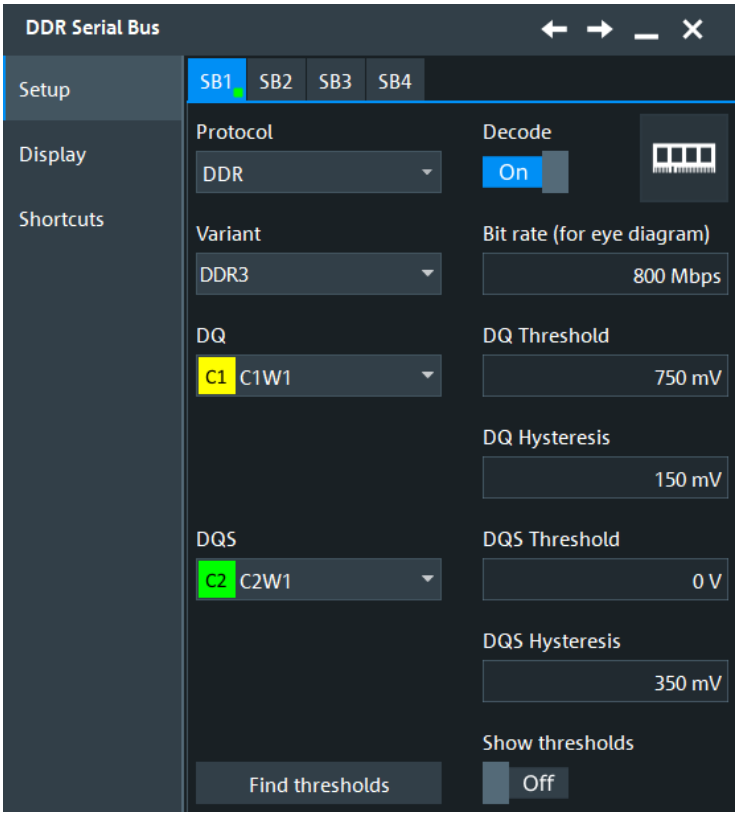
- Decode the signal.
  See:Chapter 13.27.1, "DDR configuration", on page 964
- Create and analyze an eye diagram. This helps to separate the read and write cycle.
- Together with the R&S ScopeSuite software you can perform a compliance test on the signal.
  For details refer to:
  – "DDR3 Compliance Tests, Test Procedures Manual"

13.27.1 DDR configuration

13.27.1.1 DDR configuration settings

Access: [Bus] > "DDR" > "Setup" tab

Make sure that the tab of the correct serial bus is selected.



See also: Chapter 13.1.1, "Setup - general settings", on page 485

Variant
Selects the DDR standard.

"DDR3" Requires option R&S RTO6-K91.

Remote command:
`BUS<m>:DDR:STANdard` on page 2387

DQ
Sets the channel for the data output signal.

You can select analog channels "C1" to "C4", and also active math and reference waveforms as source.

Remote command:
`BUS<m>:DDR:DATA:SOURce` on page 2388

DQS
Sets the channel for the data strobe signal.

You can select analog channels "C1" to "C4", and also active math and reference waveforms as source.

Remote command:
`BUS<m>:DDR:STRBe:SOURce` on page 2388

Threshold
Sets the threshold value for the digitization of the DQ/DQS lane.

There are two ways to set the thresholds:
- "Threshold"
  Enter individual values for each line directly in the fields.
- "Find thresholds"
  Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Remote command:
`BUS<m>:DDR:DATA:THReshold` on page 2388
`BUS<m>:DDR:STRBe:THReshold` on page 2389
`BUS<m>:SETReflevels` on page 1671

Hysteresis
Sets a value for the hysteresis of the respective lane.

Remote command:
`BUS<m>:DDR:DATA:HYSTeresis` on page 2387
`BUS<m>:DDR:STRBe:HYSTeresis` on page 2388

Bit rate (for eye diagram)
Declares the nominal bit rate of the data signal for the DDR eye diagram analysis.

See also, Range UI, Position UI.

Remote command:
`BUS<m>:DDR:BITRate` on page 2389

Show thresholds
If enabled, the threshold levels are displayed in the diagram.

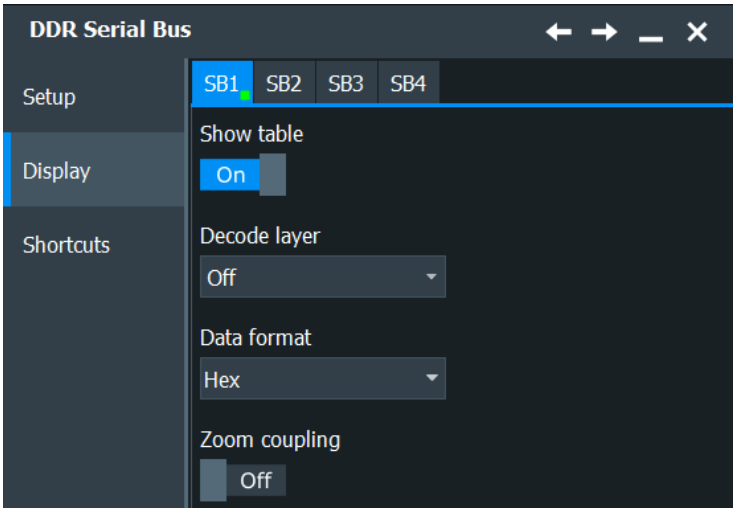
Remote command:
`BUS<m>:THReshold` on page 1671

13.27.1.2 Display settings

Access: [Bus] > "Display" tab.

For common display settings, see Chapter 13.1.3, "Display", on page 487.

To enhance the decode possibilities of the DDR protocol, you can use an additional setting in the "Display" tab: "Decode layer".



Decode layer
Selects the decode layer.

Decoding is performed in several layers, and the end results are presented in the decode table. The decode layer selects an interim step for which the decoding result is shown in the honeycomb display.

"Edges"	All edges of each lane in stage 1 decoding.
"Bits"	All bits of each lane in stage 1 decoding.
"Word"	All words of each lane in stage 1 decoding.

13.27.1.3 Configuring the DDR signals

For configuration, assign the lanes to the input channels and define the logical thresholds and the hysteresis.

1. Press the [Bus] key on the front panel.
2. Select the tab of the bus you want to set up, for example "SB1".
3. In the "Setup" tab, set "Protocol"="DDR".
4. Optionally, you can enter a "Bus label" on the "Display" tab.
5. Set the channels for the "DS" and "DQS".
6. Set the "Threshold" and "Hysteresis" values.
7. Enable "Decode".

13.27.2 Search on decoded DDR data

Using the search functionality, you can find various events in the decoded data. You can find the same events that you can trigger on, and even many more, since several event types can also be combined.

Before you can start the search, you have to configure the bus correctly and acquire decoded data.

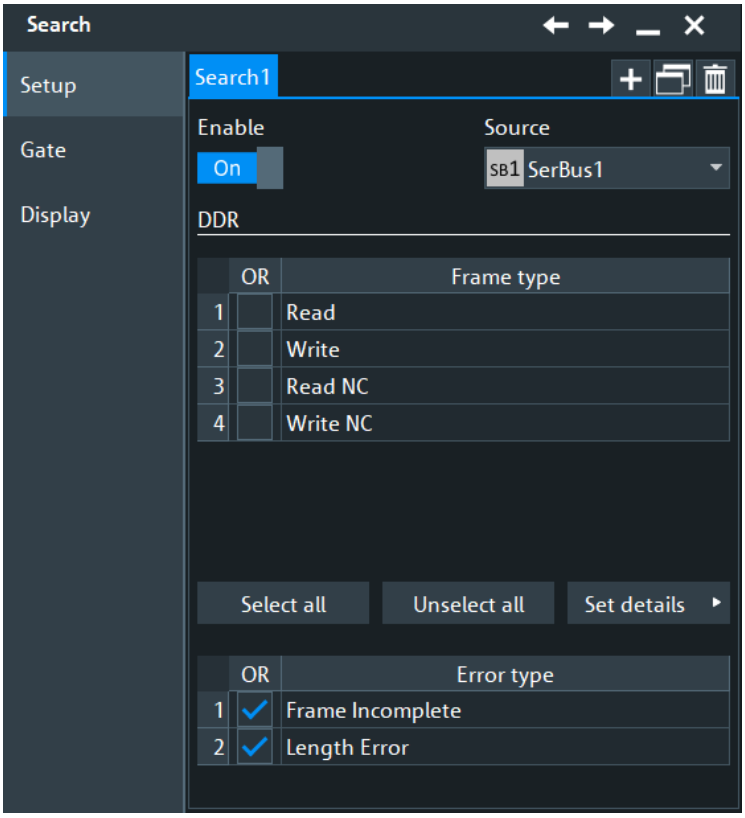
To search on decoded data, set the search source to "SerBus" for the configured protocol.

For general information on how to handle the search functionality, see Chapter 11, "Search functions", on page 419.

If you need information on how to get started with searching for DDR data, see Chapter 13.27.2.3, "Searching for DDR data", on page 971. Otherwise proceed with the DDR search setup.

13.27.2.1 DDR search setup

Access: [Bus] > "Shortcuts" tab > "Setup search"



In this dialog, you can define the search type for DDR.

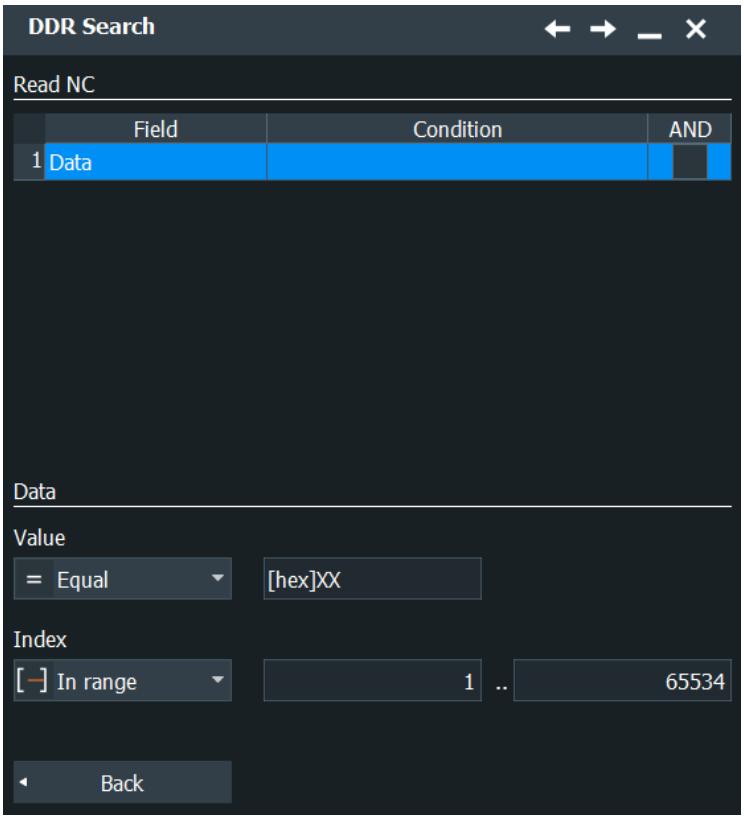
DDR search
You can select one or more frames to be searched for. Available are "Read", "Write, ", "Read NC" and "Write NC ". You can also define the data and/or index operators and values for the field, see Set details.

Remote command:
`SEARch:TRIGger:DDR:FRENable` on page 2395
`SEARch:TRIGger:DDR:FRAMe<m>:ENABle` on page 2396

Error type
You can search for "Frame Incomplete" and "Length error".

Remote command:
`SEARch:TRIGger:DDR:ERRor<m>:ENABle` on page 2394
`SEARch:TRIGger:DDR:ERENable` on page 2394

Set details
In this dialog, you can define the individual search parameters for the fields of the selected frame. Select the frame type first, and then open the "Set details" dialog.



Remote command:
`SEARch:TRIGger:DDR:FIENable` on page 2394

Value ← Set details
Sets the data value of the field to be searched for. The search data setup consists of a comparison condition and one or two data patterns.

"Condition"	Sets the comparison condition to a specific value or a range.
"Min"	Specifies the value or sets the start value of a range.
"Max"	Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
`SEARch:TRIGger:DDR:DMAX` on page 2395
`SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DMAX` on page 2395
`SEARch:TRIGger:DDR:DMIN` on page 2395

SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DMIN on page 2395
SEARch:TRIGger:DDR:DOPerator on page 2396
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DOPerator on page 2396

Index ← Set details
The search index setup consists of a comparison condition and one or two address patterns.

"Condition"	Sets the comparison condition to a specific value or a range.
"Min"	Specifies the value or sets the start value of a range.
"Max"	Sets the maximum value of a range for "Condition" = "In range"/"Out of range".

Remote command:
SEARch:TRIGger:DDR:IMAX on page 2397
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IMAX on page 2397
SEARch:TRIGger:DDR:IMIN on page 2397
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IMIN on page 2397
SEARch:TRIGger:DDR:IOPerator on page 2397
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IOPerator on page 2397

13.27.2.2 DDR search results

To get search results, "Enable" the search. You can minimize, shift or close the search dialog.

The search results are listed in the search result table and marked in the waveform by blue lines.

For an introduction to search results, see:

- Chapter 11.1.2, "Search results", on page 420
- Chapter 11.4.2, "Display zoom settings", on page 440

The results of the search are the same as for decoding, see Chapter 13.27.3, "DDR decode results", on page 971.

Remote commands:

- SEARch:RESult:DDR:FCOunt? on page 2398
- SEARch:RESult:DDR:FRAMe<m>:FLD<n>:FVAL? on page 2398
- SEARch:RESult:DDR:FRAMe<m>:FLD<n>:STARt? on page 2399
- SEARch:RESult:DDR:FRAMe<m>:FLD<n>:STATus? on page 2399
- SEARch:RESult:DDR:FRAMe<m>:FLD<n>:STOP? on page 2400
- SEARch:RESult:DDR:FRAMe<m>:FLD<n>:VAL? on page 2400
- SEARch:RESult:DDR:FRAMe<m>:NWRDs? on page 2400
- SEARch:RESult:DDR:FRAMe<m>:STARt? on page 2401
- SEARch:RESult:DDR:FRAMe<m>:STATe? on page 2401
- SEARch:RESult:DDR:FRAMe<m>:STOP? on page 2401

- `SEARch:RESult:DDR:FRAMe<m>:TYPE?` on page 2402

13.27.2.3 Searching for DDR data

Prerequisite: A serial bus is configured for the DDR signal to be decoded and analyzed.

The search for events is set up in the following way:

1. Press the [Bus] key and select the "Shortcuts" tab.
2. Tap "Setup search".
3. If the dialog box does not contain a search entry, tap the "Add" icon to create one, as described in "To create a user-defined search" on page 434.
4. Tap "Source" and select the serial bus that is set to DDR (e.g. "SB1", unless already selected).

   The search dialog for DDR protocol analysis is opened.
5. Specify search criteria according to Chapter 13.27.2.1, "DDR search setup", on page 968.
6. To acquire a waveform, press [Single].

   The R&S RTO6 performs a DDR decode according to the thresholds and protocol settings of the associated serial bus source (here in our example SB1).
7. To start searching the acquired waveform for specific events, tap "Enable" in the search setup dialog:

   The R&S RTO6 displays the "Search Results" box that lists the detected events.

For information on how to configure the search results presentation and how to navigate the search results, see also "To display search zoom windows" on page 442.

13.27.3 DDR decode results

When the configuration of the serial bus is complete, the signal can be decoded:

1. In the "Protocol" dialog > "Setup" tab, enable "Decode".

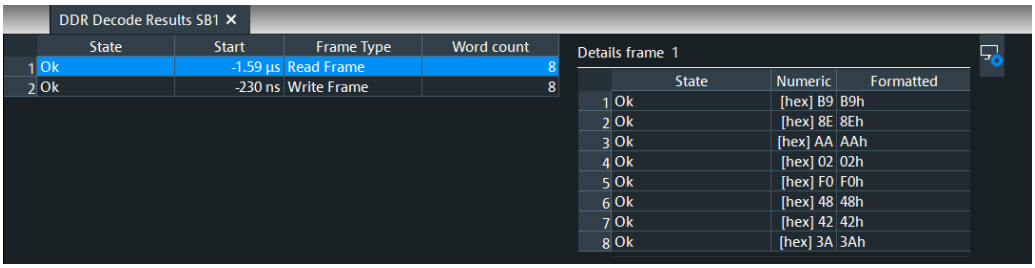
2. In the "Protocol" dialog > "Display" tab, select additional result display settings: "Show decode table" and "Show binary signals".

For a description of the display settings, see also Chapter 13.1.3, "Display", on page 487.

The instrument captures and decodes the signal according to the standard definition and the configuration settings.

The color-coding of the various protocol sections and errors simplifies the interpretation of the visual display. The decode information condenses or expands, depending on the horizontal scale. Various data formats are available to show the result values.

Decode Results Table



The results are shown in two tables:

- "Decode results": contains information about all decoded frames
- "Details frame": contains more detailed information about the selected frame in the "Decode results" table.

Table 13-51: Content of the Decode result table

Column	Description
Frame	Frame count
State	Overall state of the frame indicating, for example, if the frame is valid or invalid
Start	Time of frame start in relation to the trigger point
Frame Type	Frame identifier specifying the data or control frame name
Word count	Total number of words in the frame

Table 13-52: Content of the Details frame table

Column	Description
State	State of the field
Numeric	Hexadecimal value of the field
Formatted	Formatted content of the field

Remote commands

Remote commands are described in Chapter 24.17.28.2, "Decode results", on page 2389.

13.27.4 Eye diagram setup

This chapter describes the configuration of eye diagrams.

The configuration of an eye diagram mainly consists of the selection and setup of the input source and timing reference source. You can also adjust display settings and restrict the segments that contribute to the eye generation by various means.

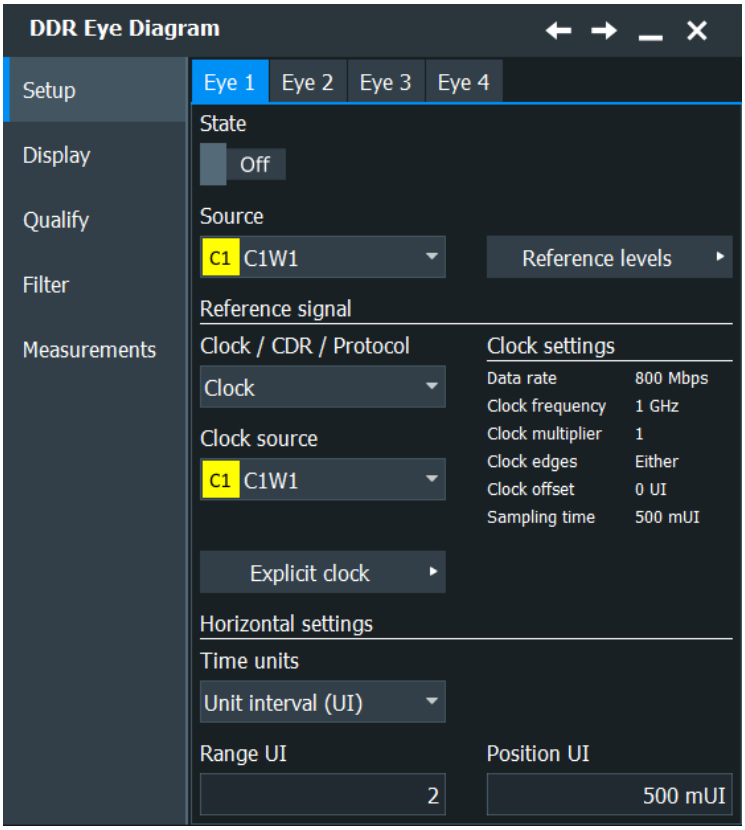
You can configure and analyze up to four independent eye diagrams in parallel.

The "Eye Diagram" dialog box provides four tabs with all settings that are required to configure eye diagrams.

13.27.4.1 Basic setup

Access: [App Cockpit]> "Analysis" > "Eye Analysis" > "Setup" tab

In the "Setup" tab, you configure the source, the timing reference, and horizontal scale settings of the eye diagram. The vertical scale of the eye diagram is coupled to the vertical scale of the data source.



You can configure and analyze up to four independent eye diagrams in parallel (vertical subtabs Eye 1 to Eye 4). Make sure to select the correct eye diagram subtab on the left before you adjust the settings.

State....974
Source....974
Reference signal....974
└ Clock / CDR / Protocol....974
└ Clock....974
└ CDR....975
└ Protocol....975
Horizontal settings....975
└ Time units....975
└ Range, Position....975
└ Range UI, Position UI....976
Explicit clock....976
└ Vertical reference level....976
└ Clock frequency....977

└ Estimate clock frequency....977
└ Clock multiplier....977
└ Clock edge....977
└ Data rate....977
└ Clock offset....977
└ Estimate clock offset....977
└ Sampling time....978

State
Activates or deactivates the eye diagram.

Remote command:
`EYE<m>[:STATe]` on page 2403

Source
Selects the waveform from which the eye diagram is generated (data source).

Select the same channel for the DDR serial bus protocol configuration "DQ" data line and the input for the eye diagram.

Remote command:
`EYE<m>:SOURce` on page 2403

Reference signal
In this section you can define the reference signal used for the eye diagram.

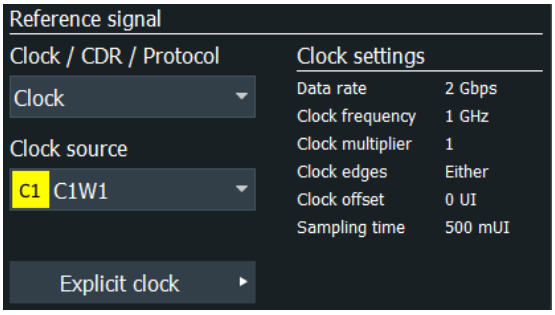
Clock / CDR / Protocol ← Reference signal
Selects the type of waveform that is used to obtain the timing information required to slice the data source waveform.

Remote command:
`EYE<m>:TIMReference[:SOURce]` on page 2407

Clock ← Reference signal
If a clock signal is available, the instrument can use it directly to set the segmentation timestamps. The signal has to be provided on an additional channel.

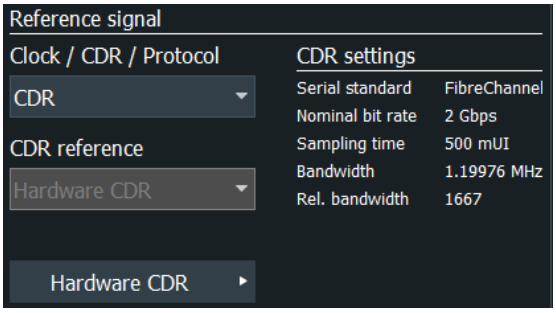
With "Clock source" you can select the source of the clock signal.

See also Explicit clock.



Remote command:
`EYE<m>:TIMReference:CLOCk:SOURce` on page 2407

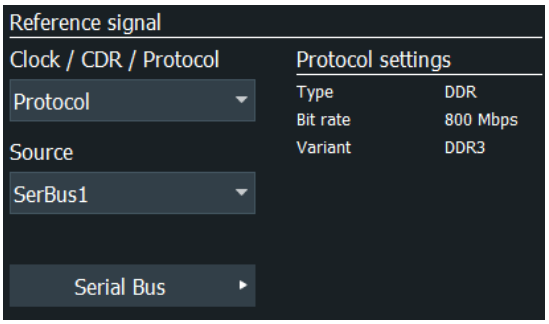
CDR ← Reference signal
Uses a hardware clock data recovery.



Protocol ← Reference signal
If "Protocol" is selected as the reference type signal, the DDR source signal settings are used for the timing information.

You can select the "Source" of the protocol signal. You can only select the protocols, that are enabled.

Also some protocol-specific settings are displayed: "Type", "Bit rate", "Variant".



Remote command:
`EYE<m>:TIMReference:PROTocol:SELect` on page 2407

Horizontal settings
Defines the horizontal settings.

Time units ← Horizontal settings
Sets the method to define the horizontal scale of the eye diagram: as absolute time, or in unit intervals. You can use unit interval, if the nominal frequency of the clock signal is known.

Indirectly, the horizontal scale determines the size of one waveform slice. The waveform slice must be fully contained in the waveform acquisition to contribute to the eye diagram.

Remote command:
`EYE<m>:HORizontal[:UNIT]` on page 2403

Range, Position ← Horizontal settings
Available for Time units = "Absolute time"

Define the horizontal scale in absolute time.

"Range" Time range that is covered by the eye diagram.

"Position" Place of the zero point in the diagram, in seconds. The zero point is the alignment point on which the slice timestamps are superimposed.

Remote command:
`EYE<m>:HORizontal:ABSolute:RANGe` on page 2403
`EYE<m>:HORizontal:ABSolute:POSition` on page 2404

Range UI, Position UI ← Horizontal settings
Available for Time units = "Unit interval (UI)"

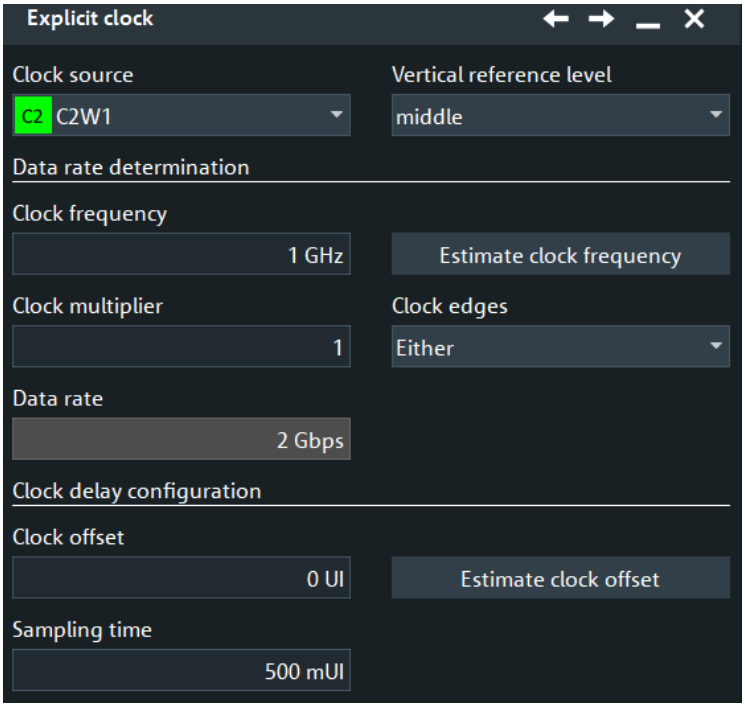
Define the horizontal scale in unit intervals. A unit interval typically corresponds to a bit in the input data signal.

"Range UI" Number of unit intervals that are visible in the eye diagram.

"Position UI" Place of the zero point in the diagram, in unit intervals. The zero point is the alignment point on which the segment timestamps are superimposed.

Remote command:
`EYE<m>:HORizontal:UINTerval:RANGe` on page 2404
`EYE<m>:HORizontal:UINTerval:POSition` on page 2404

Explicit clock
Opens a dialog to define the setting of the explicit clock for "Clock / CDR / Protocol" = "Clock".



Vertical reference level ← Explicit clock
Sets the reference level to define the timestamps for slicing the data waveform. A timestamp is set each time when the selected slope of the timing waveform crosses the selected reference level.

Remote command:
`EYE<m>:TIMReference:CLOCk:LEVel` on page 2406

Clock frequency ← Explicit clock
Sets the clock frequency of the explicit clock signal. If this value is unknown, you can use "Estimate clock frequency".

Remote command:
`EYE<m>:TIMReference:CLOCk:FREQuency` on page 2405

Estimate clock frequency ← Explicit clock
Calculates an estimate value for the clock frequency.

Remote command:
`EYE<m>:TIMReference:CLOCk:FREStimation` on page 2406

Clock multiplier ← Explicit clock
Sets a value for the clock multiplier, the ratio of an internal clock rate to the externally supplied clock.

Remote command:
`EYE<m>:TIMReference:CLOCk:FACTor` on page 2405

Clock edge ← Explicit clock
The clock edge sets the point in time when the state of the data signal is checked:

"Positive" Rising edge, a positive voltage change.

"Negative" Falling edge, a negative voltage change.

"Either" Both the rising and the falling edge.

Remote command:
`EYE<m>:TIMReference:CLOCk:EDGE` on page 2405

Data rate ← Explicit clock
Displays the calculated data rate. It is calculated as follows:

*"Data rate" = "Clock frequency" * "Clock multiplier"*X*

Where *X* is:

- 1 for "Clock edge"= "Positive" or "Negative"
- 2 for "Clock edge"= "Either"

Remote command:
`ADVJitter<m>:SIGNal:DATRate?` on page 2488

Clock offset ← Explicit clock
Sets a clock offset. If this value is unknown, you can use "Estimate clock offset".

Remote command:
`EYE<m>:TIMReference:CLOCk:OFFSet` on page 2406

Estimate clock offset ← Explicit clock
Calculates an estimate value for the clock offset.

Remote command:
`EYE<m>:TIMReference:CLOCk:OFEStimation` on page 2406

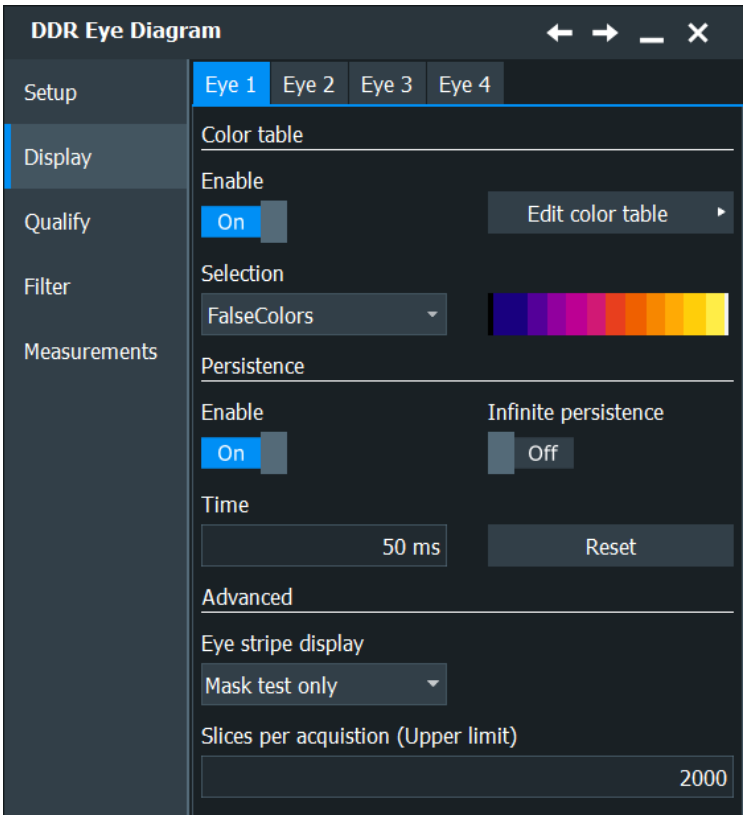
Sampling time ← Explicit clock
Sets a sampling time for the explicit clock signal, an offset for the clock edge in relation to the bit start. The clock edge sets the beginning of the unit interval. The sampling time is a number between 0 and 1. Value 0 sets the clock edge to the beginning of the bit period; value 0.5 sets the clock edge to the middle of the bit period.

Remote command:
`EYE<m>:TIMReference:CLOCk:SAMTime` on page 2407

13.27.4.2 Display settings

Access: [App Cockpit]> "Analysis" > "DDR Eye" > "Display" tab

In the "Display" tab, you define the appearance of the individual eye diagrams.



You can configure and analyze up to four independent eye diagrams in parallel (vertical subtabs Eye 1 to Eye 4). Make sure to select the correct eye diagram subtab on the left before you adjust the settings.

Color table
Like any other waveform, you can show the eye waveform in solid color, or apply a color table.

For a description of the color table settings, see Chapter 4.3.2, "Color tables", on page 76.

Persistence
Each eye diagram has its specific persistence settings, independent of the general waveform persistence.

Enable ← Persistence
If enabled, each new data point in the diagram area remains on the screen for the duration that is defined using Time, or as long as Infinite persistence is selected.

If disabled, the waveform points are displayed only for the current acquisition.

Remote command:
`EYE<m>:DISPlay:PERSistence:STATe` on page 2409

Infinite persistence ← Persistence
If infinite persistence is enabled, each new waveform point remains on the screen until this option is disabled. Use infinite persistence to display rare events in the signal.

Remote command:
`EYE<m>:DISPlay:PERSistence:INFinite` on page 2409

Time ← Persistence
Sets a time factor that controls how long the waveform points fade away from the display. Thus, the R&S RTO6 emulates the persistence of analog phosphor screens.

Remote command:
`EYE<m>:DISPlay:PERSistence:TIME` on page 2409

Reset ← Persistence
Resets the display, removing persistent all waveform points.

Remote command:
`EYE<m>:DISPlay:PERSistence:RESet` on page 2409

Eye stripe display
Defines when the eye stripe is shown: always, never, or during mask tests.

The eye stripe is shown at the bottom of the main source waveform. It shows the part of the waveform that contributes to the eye. When a mask test is performed on the eye diagram, the eye stripe shows the position of the mask violations in red color.

Remote command:
`EYE<m>:DISPlay:MODE` on page 2408

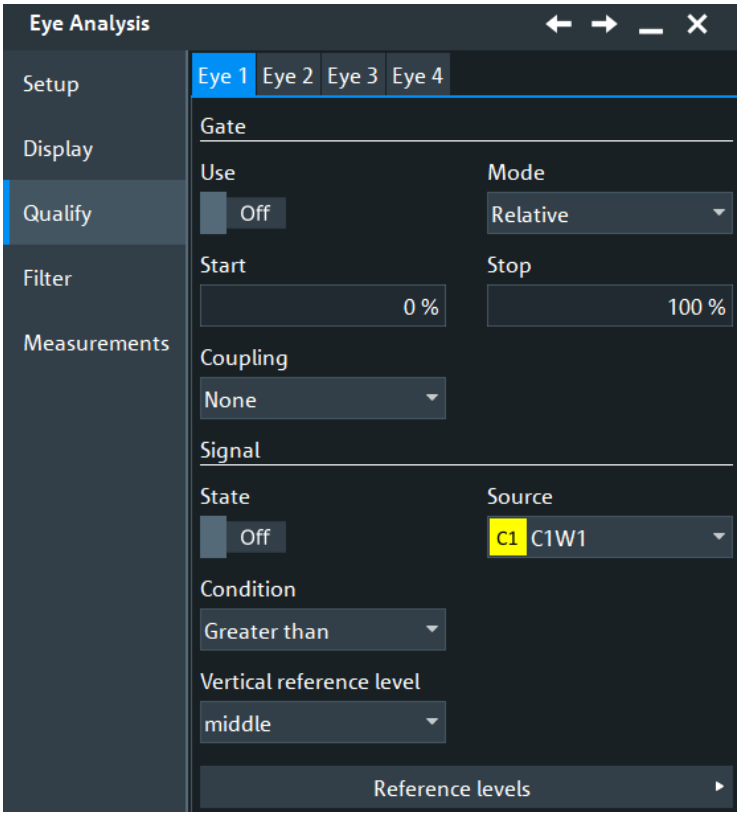
Slices per acquisition
Sets the number of waveform slices for a single acquisition.

Remote command:
`EYE<m>:MSLices` on page 2408

13.27.4.3 Qualify settings

Access: [App Cockpit]> "Analysis" > "DDR Eye" > "Qualify" tab

Using the qualification settings, you can restrict the slices that contribute to the eye diagram. The restriction can be in horizontal direction (gate), or in vertical direction using the reference level.



You can configure and analyze up to four independent eye diagrams in parallel (vertical subtabs Eye 1 to Eye 4). Make sure to select the correct eye diagram subtab on the left before you adjust the settings.

Gate
Restricts the contributing slices in horizontal direction. Only the timestamps within the defined horizontal interval qualify for eye generation.

To apply the gate settings, enable the "State".

The gate definition is the same as gate definition for measurements, but the gates are independent. You can define the gate in one of these ways:

- Enter absolute time values for gate start and gate end: "Coupling mode = None" and "Mode = Absolute". Set "Start" and "Stop" time.
- Enter relative values for gate start and gate end: "Coupling mode = None" and "Mode = Relative". Set "Relative start" and "Relative stop" in percent.
- If a cursor measurement is active, you can use the vertical cursor lines as gate limits: "Coupling mode = Cursor". Select the cursor set.
- If a zoom is active, you can use the zoom limits as gate: "Coupling mode = Zoom". Select the zoom.

Remote command:
`EYE<m>:QUALify:GATE[:STATe]` on page 2410
`EYE<m>:QUALify:GATE:COUPling` on page 2410
`EYE<m>:QUALify:GATE:MODE` on page 2410
`EYE<m>:QUALify:GATE:ABSolute:STARt` on page 2411
`EYE<m>:QUALify:GATE:ABSolute:STOP` on page 2411
`EYE<m>:QUALify:GATE:RELative:STARt` on page 2411
`EYE<m>:QUALify:GATE:RELative:STOP` on page 2411
`EYE<m>:QUALify:GATE:CURSor` on page 2412
`EYE<m>:QUALify:GATE:ZDIagram` on page 2412

Signal
Sets a waveform-based qualification for the eye generation. The voltage level of a waveform is compared with a threshold, and only if this condition is fulfilled, the slices contribute to the eye generation.

State ← Signal
Applies the signal qualification.

Remote command:
`EYE<m>:QUALify:SIGNal[:STATe]` on page 2412

Source ← Signal
Selects the waveform to be compared.

Remote command:
`EYE<m>:QUALify:SIGNal:SOURce` on page 2413

Condition ← Signal
Selects the comparison condition for the waveform.

Remote command:
`EYE<m>:QUALify:SIGNal:CONDition` on page 2413

Vertical reference level ← Signal
Selects the threshold. You can use the lower, middle, or upper reference level. These levels are the reference levels that are defined for measurements.

See Chapter 8.2.4, "Reference levels", on page 307.

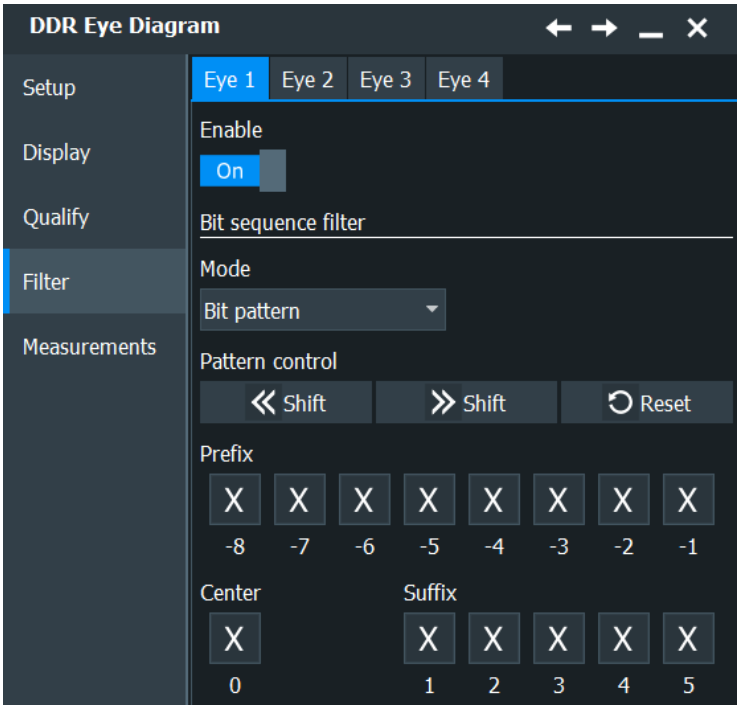
Remote command:
`EYE<m>:QUALify:SIGNal:LSELect` on page 2413

Reference levels
Opens the reference level dialog. For details, see Chapter 8.2.4, "Reference levels", on page 307.

13.27.4.4 Filter settings

Access: [App Cockpit]> "Analysis" > "DDR Eye" > "Filter" tab

Filter settings allow you to select waveform slices for eye generation dependent on the bit status.



You can configure and analyze up to four independent eye diagrams in parallel (vertical subtabs Eye 1 to Eye 4). Make sure to select the correct eye diagram subtab on the left before you adjust the settings.

Enable
Applies the filter settings.

Remote command:
`EYE<m>:FILTer:STATe` on page 2414

Mode
Selects which bits contribute to the eye generation.

"All bits"	All bits contribute to the eye diagram, same as "State" = off.
"Level transition"	Only bits after a signal transition contribute (0 to 1, or 1 to 0).
"Constant level"	Only bits without a preceding signal transition contribute (0 to 0, or 1 to 1).
"Bit pattern"	Only bits contribute that match the defined pattern before and after the center eye. Use the bit pattern, for example, to measure the effect of data-dependent jitter on the eye diagram.

Remote command:
`EYE<m>:FILTer:BPATtern:MODE` on page 2414

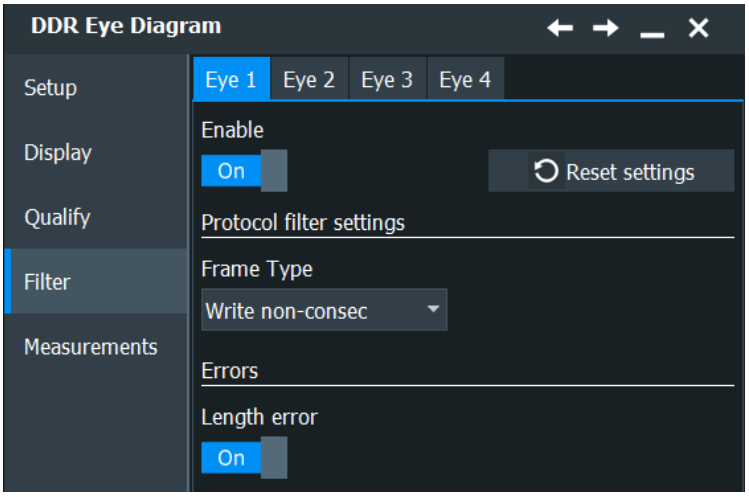
Pattern
Defines the bit pattern if "Mode" = "Bit pattern". You can set up to 8 bits before and 5 bits after the center eye (reference bit).

Remote command:
`EYE<m>:FILTer:BPATtern:PREFix<1..8>` on page 2415
`EYE<m>:FILTer:BPATtern:CENTer` on page 2415
`EYE<m>:FILTer:BPATtern:SUFFix<1..5>` on page 2415

Pattern control
Shift the pattern bit by bit to the left or to the right, and reset to the default (X for all bits).

Remote command:
`EYE<m>:FILTer:BPATtern:SLEFt` on page 2415
`EYE<m>:FILTer:BPATtern:SRIGht` on page 2415
`EYE<m>:FILTer:BPATtern:RESet` on page 2415

Protocol filter settings
This setting is only available, if the Source is set to "SBx" and the "DDR" serial bus analysis is enabled for the same bus.



Reset settings ← Protocol filter settings
Resets the current filter settings.

Frame Type ← Protocol filter settings
Selects which frame types contribute to the eye diagram, write frames, read frames or any of them.

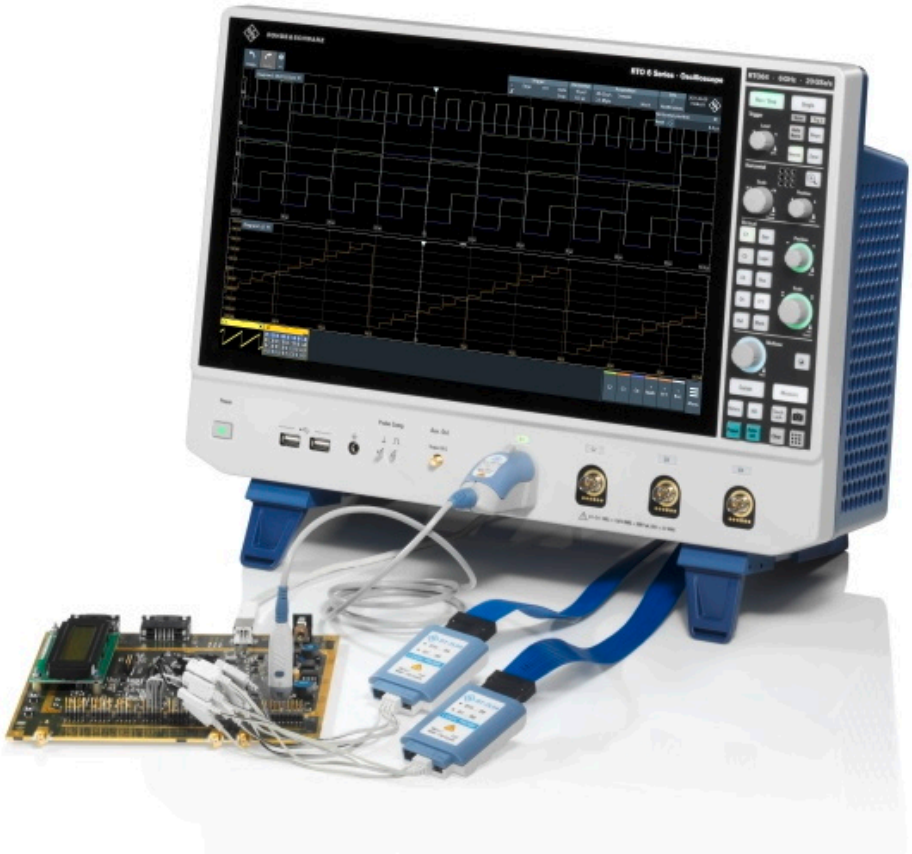
Remote command:
`EYE<m>:FILTer:DDR:FTYPe` on page 2415

Length Error ← Protocol filter settings
If enabled, the length errors contribute to the eye diagram.

Remote command:
`EYE<m>:FILTer:DDR:LENerror` on page 2416

14 Mixed signal option (MSO, R&S RTO6-B1)

The Mixed Signal Option R&S RTO6-B1 adds logic analyzer functions to the classical oscilloscope functions. Using the MSO option, you can analyze and debug embedded systems with mixed-signal designs that use analog signals and correlated digital signals simultaneously.



The Mixed Signal Option provides 16 digital channels grouped in two logic probes (pods) with 8 channels each. The instrument ensures that analog and digital waveforms are time-aligned and synchronized so that critical timing interactions between analog and digital signals can be displayed and tested. The automatic alignment compensates the skew between the probe connectors of the analog channels and the probe boxes of the digital channels.

Logic probes provided by Rohde & Schwarz are recognized by the instrument. The information on each recognized probe is shown under "Menu" "Settings" > "Maintenance" > "Hardware" > "Advanced" > "Logic Probes".

14.1 Digital channels and parallel buses

Each digital channel can be displayed on the screen and used as trigger source. Digital channels can be grouped and displayed as a parallel bus. Up to four parallel buses can be configured; and two bus types are supported: clocked bus and unclocked bus. The clocked bus is available only on parallel bus 1 and 2. Each digital channel can be assigned to one *active* parallel bus only, the instrument disables conflicting buses automatically.

You can display each bus and use it as trigger source, as well. For each active parallel bus, the corresponding signal icon appears on the signal bar and indicates the assigned digital channels. Individual digital channels do not have a signal icon.

If one or more parallel buses are active, the roll mode is not available.

14.1.1 Parallel buses - configuration

Access: "Menu" > "Logic" > "Setup" tab

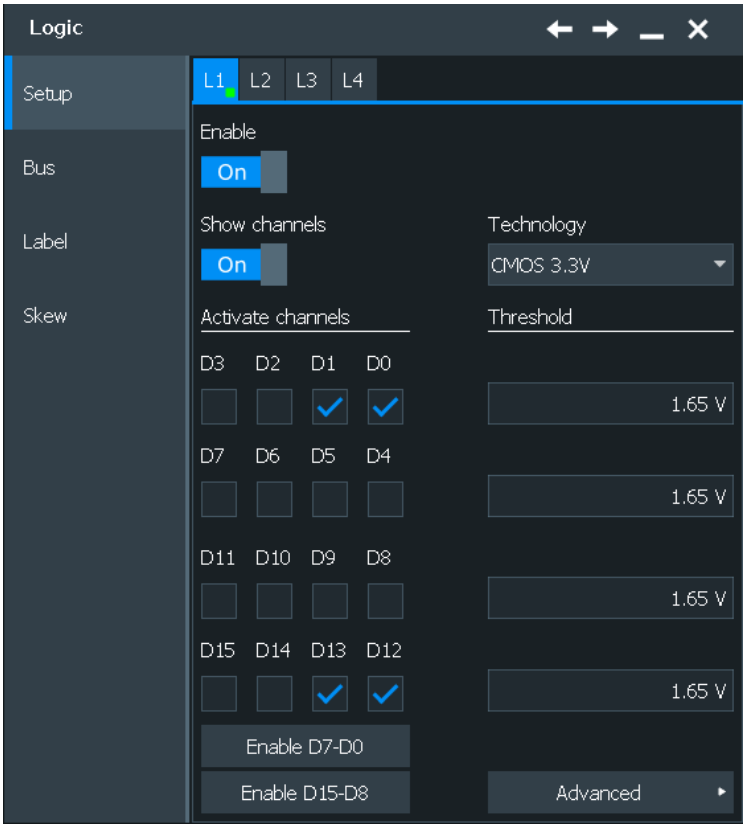
Digital channels can be displayed individually, and they can be grouped and displayed as a parallel bus. You can configure and enable up to 4 parallel buses. Each digital channel can be assigned to one *active* parallel bus only, the instrument disables conflicting buses automatically.

For clocked buses, you can display the decoded data in a result box.

If you have configured several parallel buses and you want to modify the settings, make sure that the tab of the correct bus is selected, and disable the bus before you change the settings.

14.1.1.1 Setup

Access: "Menu" > "Logic" > "Setup" tab



Enable bus
Enables the selected parallel bus. The corresponding signal icon appears on the signal bar.

If another *active* bus already uses the same digital channel, the instrument disables the other bus and shows a message.

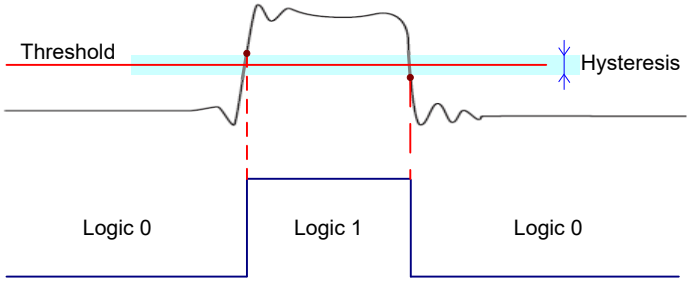
Remote command:
`BUS<m>:PARallel:STATe` on page 2424

Show channels
If enabled, the selected digital channels are shown in the diagram. Each channel is displayed as a logic signal.

Remote command:
`BUS<m>:PARallel:DISPlay:SHDI` on page 2428

Technology, Threshold
Sets the logical threshold. For each sample, the instrument compares the input voltage with the threshold value. If the input voltage is above the threshold, the signal state "1" is stored. Otherwise, the signal state "0" is stored if the input voltage is below the threshold.

To avoid the change of signal states due to noise, a hysteresis is considered.



By default, same threshold and hysteresis value are used for all digital channels and all parallel buses: "Couple thresholds" is enabled.

You can also set different thresholds for the individual channel groups. As long as the buses are disabled, you can set different thresholds for each bus. Active buses use the same threshold and hysteresis values, the settings of the last activated bus take effect.

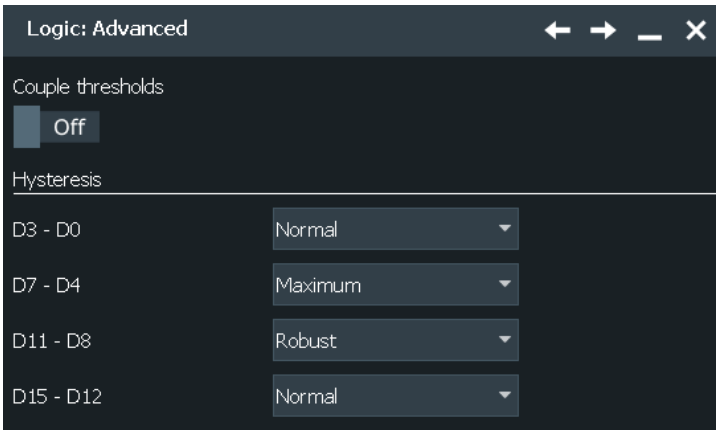
The range of threshold levels and the minimum voltage swing is given in the data sheet.

"Threshold" Enter the value directly in the field.

"Technology" Selects the threshold voltage for various types of integrated circuits from a list and applies it to all digital channels. The value is set to "Manual" if a user-defined threshold was entered directly.

Remote command:
`BUS<m>:PARallel:TECHnology` on page 2425 (all buses)
`DIGital<m>:TECHnology` on page 2421 (bus1)
`BUS<m>:PARallel:THReshold<n>` on page 2424 (all buses)
`DIGital<m>:THReshold` on page 2421 (bus1)

Advanced
Opens a dialog for advanced threshold and hysteresis settings. You can define them additional to the "Threshold" and "Technology".



Couple thresholds ← Advanced
Sets the threshold and the hysteresis for all digital channels and all buses to the same value.

Remote command:
`BUS<m>:PARallel:THCoupling` on page 2425 (all buses)
`DIGital<m>:THCoupling` on page 2422 (bus1)

Hysteresis ← Advanced
Defines the size of the hysteresis for the respective channels.

"Normal"	The instrument sets a small value suitable for the signal and its settings. Use this setting for clean signals.
"Maximum"	The instrument sets the maximum value that is possible and useful for the signal and its settings. Use this setting for noisy signals.
"Robust"	Sets different hysteresis values for falling and rising edges to avoid an undefined state of the trigger system. Use this setting for very noisy signals.

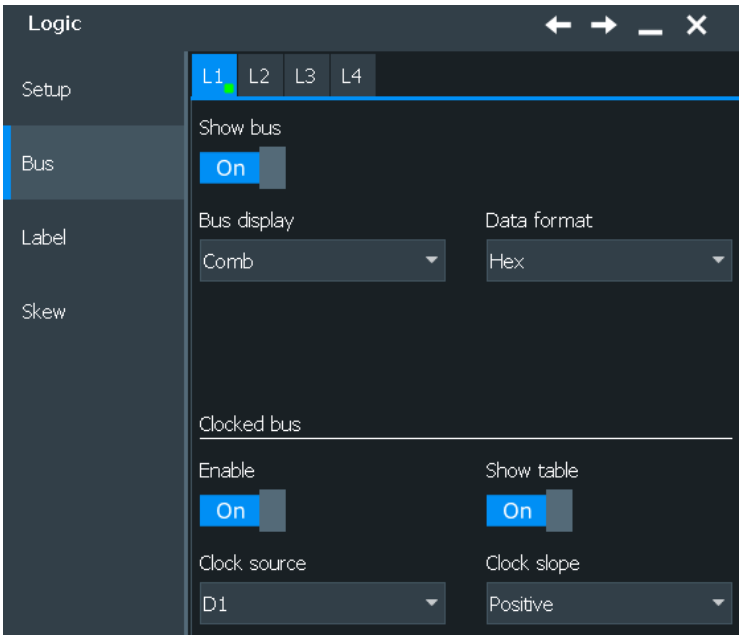
Remote command:
`BUS<m>:PARallel:HYSTeresis<n>` on page 2426 (all buses)
`DIGital<m>:HYSTeresis` on page 2422 (bus1)

Activate channels
Selects the state of the respective bus channel.

Enable/Disable D7-D0, D15-D18
The buttons select or deselect all digital channels of a pod at once.

14.1.1.2 Bus

Access: "Menu" > "Logic" > "Bus" tab



Show bus
If enabled, the resulting bus signal and bus values are displayed in the diagram. Select the presentation type for the bus signal with "Bus display" on page 989.

Remote command:
`BUS<m>:PARallel:DISPlay:SHBU` on page 2428

Bus display
Defines how the parallel bus is displayed.

"Comb"	Displays the decoded bus signal with bus values. When at least one digital channel changes its value, the bus value changes too.
"Analog"	Displays the bus values as signal amplitudes, similar to an analog waveform. Thus, a quasi-analog waveform is created.

Remote command:
`BUS<m>:PARallel:DISPlay:BTYP` on page 2428

Data format
Sets the data format of bus values, which are displayed in the decode table and on the comb bus display. Available formats are: Hex, ASCII, octal, binary, signed, and unsigned.

Signed and Unsigned are integer data types with maximum 16-bit length. Unsigned is used for positive integers. Signed is used for positive and negative integers.

If the target file format is BIN, you can save only signed and unsigned binary data. The data format "Signed" writes signed data; all other formats are saved as unsigned binary data.

Remote command:
`BUS<m>:PARallel:DATA:FORMat` on page 2427

Clocked bus
If a bus is a clocked bus, one of the digital channels serves as clock of the bus.

For an unclocked bus, the logical state of the bus is determined for each sample. For a clocked bus, the logical state is determined only at the specified clock edges.

The settings are only available for "Bus1" and "Bus2".

Enable ← Clocked bus
Enable this option for a clocked bus.

Remote command:
`BUS<m>:PARallel:CLON` on page 2429

Show table ← Clocked bus
The decode table is only available for clocked buses to check the data words.

If enabled, a results box opens with decoded values of the bus signal and its time. Each clock edge corresponds to one row in the table.

Remote command:
`BUS<m>:PARallel:DECTable<n>:SHOW` on page 2430
`BUS<m>:PARallel:DECTable<n>:COUNt?` on page 2430
`BUS<m>:PARallel:DECTable<n>:DATA?` on page 2430

Clock source ← Clocked bus
Selects the digital channel used as clock.

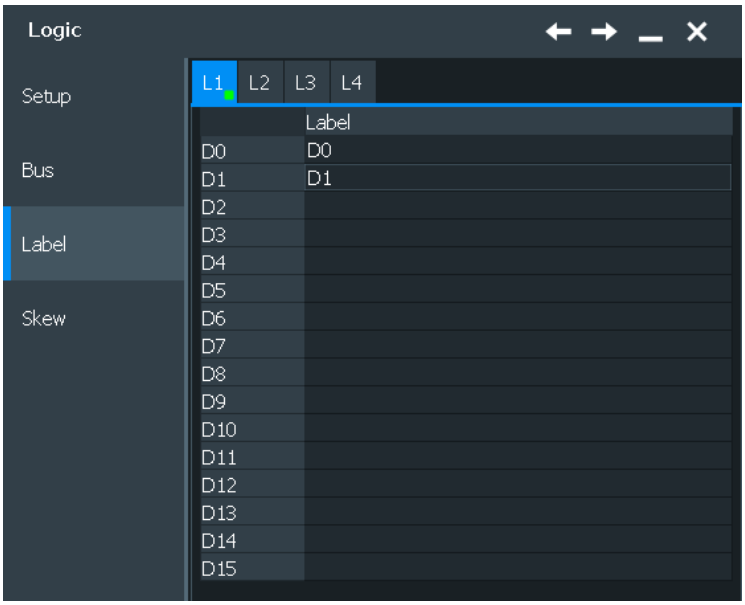
Remote command:
`BUS<m>:PARallel:CLOCk` on page 2429

Clock slope ← Clocked bus
Selects the slope of the clock signal at which all digital channels of the bus are analyzed.

Remote command:
`BUS<m>:PARallel:CLSLope` on page 2429

14.1.1.3 Label settings

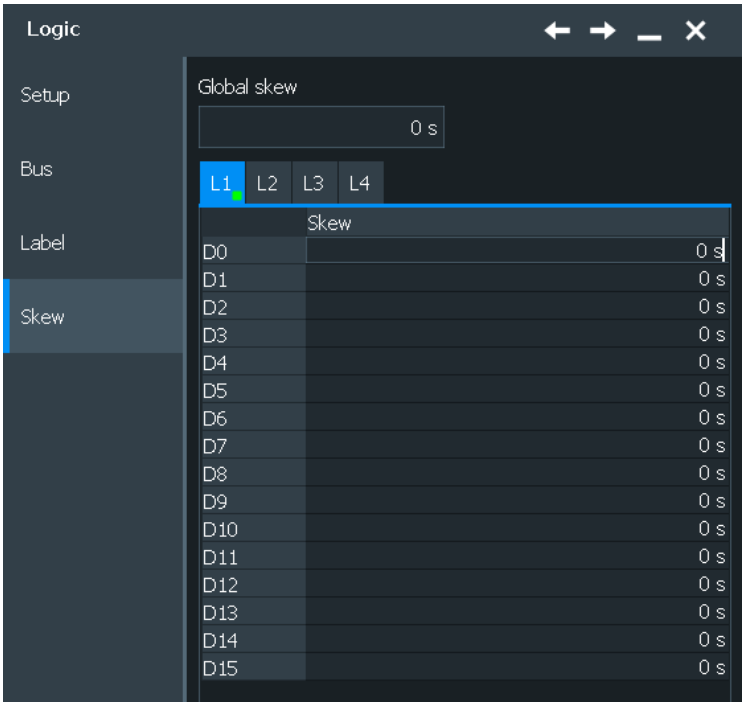
Access: "Menu" > "Logic" > "Label" tab



In this tab, you can enter a name for each digital channel. The name is displayed in the diagram.

14.1.1.4 Skew settings

Access: "Menu" > "Logic" > "Skew" tab



Global skew
Sets a general delay for all digital channels. The resulting skew of a digital channel is the sum of the general "Global skew" and the individual "Skew".

Remote command:
`BUS<m>:PARallel:DESoffset` on page 2427

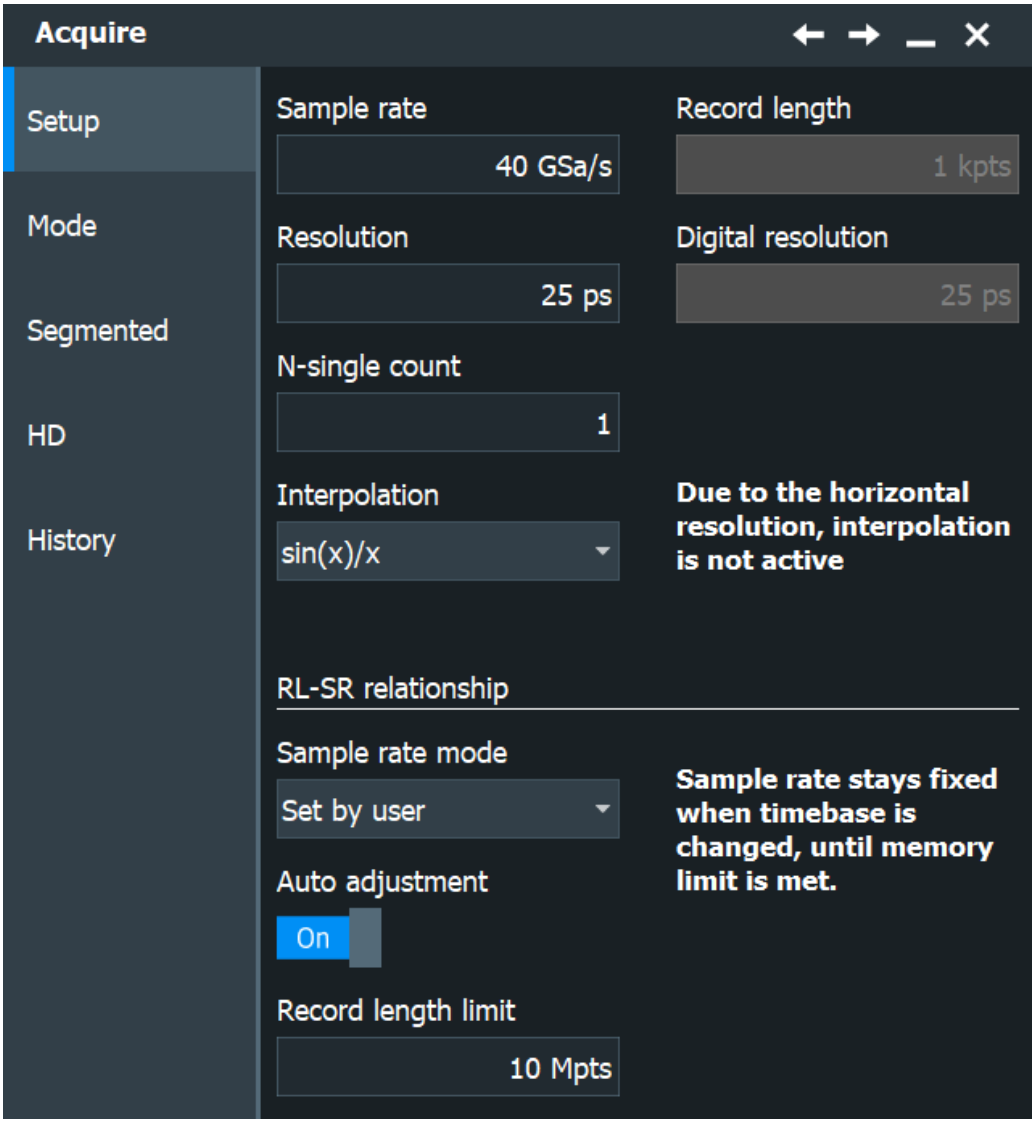
Skew
Sets an individual delay for each digital channel to time-align it with other digital channels. The skew value compensates delays that are known from the circuit specifics or caused by the different length of cables. The skew between the probe boxes of the digital channels and the probe connectors of the analog channels is automatically aligned by the instrument.

Remote command:
`BUS<m>:PARallel:BIT<n>:DESKew` on page 2426 (all buses)
`DIGital<m>:DESKew` on page 2423 (Bus1)

14.1.2 Digital resolution

Access: "Menu" > "Acquire"

If an MSO option is installed and at least one digital channel is active, additional information appears on the "Setup" tab of the "Acquire" dialog box.



Dig. resolution
Shows the current digital resolution of the digital channels. The maximum digital record length is always 200 MSa per digital channel. This number is independent of the installed memory.

Remote command:
`ACQuire:DRESolution?` on page 2430

14.1.3 Using digital probes

NOTICE

Ensuring accurate measurement results

The mixed-signal-option (MSO, R&S RTO6-B1) with connected probe leads is considered as a test probe, according to EN 61326-2-1, clause 5.2.4.101. Therefore, the measurements are sensitive to electromagnetic interference. Consider additional shielding methods to avoid interference.

Consider the following guidelines for good probing practices:

- The ground lead from each digital channel group (D15–D8 and D7–D0) should be attached to the ground of the DUT if any channel within the group is being used for data capture. The ground lead improves signal fidelity to the oscilloscope, ensuring accurate measurements.
- For high-speed timing measurements (rise time < 3 ns), each digital channel probe should use its own ground.

1. Connect the digital probe cable to any of the MSO connectors on the rear panel of the instrument as shown on the Documentation Card delivered with the digital probe.
2. Connect the ground lead on each set of channels (each pod) with a probe grabber.
3. Connect a grabber to one of the probes leads.
4. Connect the grabber to a node in the circuit that you want to test.
5. For high-speed signals, connect a ground lead to the probe lead. Connect the ground lead to ground in the device under test.
6. Repeat these steps until you have connected all points of interest.

14.1.4 Configuring digital channels and parallel buses

The configuration of a parallel bus includes the selection and setup of the digital channels, the configuration of the bus display, and, if necessary, the clock configuration.

For a detailed description of the settings, see Chapter 14.1.1, "Parallel buses - configuration", on page 985.

1. Open "Menu" > "Logic" > "Setup".
2. Enable the digital channels to be displayed and included in the bus.
   To enable or disable all channels of a pod at once, tap "Enable D7-D0" or "Enable D15-D8".

   Enabling one or more channels also enables the display of the signals - "Show channels", and enables the parallel bus. If another active bus already uses the same digital channel, the instrument disables this bus and shows a message.

The digital signals are shown in the diagram, and the signal icon of the parallel bus appears on the signal bar. Using this bus icon, you can minimize, arrange, and switch off the bus together with its channels in the same way as you do with any waveform.

3. Set the logical thresholds as described in Chapter 14.1.5, "Setting the logical thresholds", on page 994.
4. If the bus has a clock signal, open the "Bus" tab.
   Enable "Clocked bus" and select the "Clock source" and "Clock slope".
5. If necessary, enter a "Label" for each digital channel in the "Label" tab.
6. If necessary, enter a "Skew" value to time-align the channel in the "Skew" tab.

   Now the configuration of the parallel bus is completed.

14.1.5 Setting the logical thresholds

For a detailed description of the settings, see Chapter 14.1.1.1, "Setup", on page 985.
Threshold settings are the same for all *active* parallel buses.

1. Open "Menu" > "Logic" > "Setup".
2. To set the thresholds, use one of the following ways:
   a) Use the same value for all digital channels and all parallel buses:
      - In the "Logic: Advanced" dialog, enable "Coupe thresholds".
      - Open the "Setup" tab. To set the threshold, either select a predefined "Technology" value or enter a user-defined value.
   b) Set different thresholds for the individual channel groups:
      - In the "Logic: Advanced" dialog, disable "Coupe thresholds".
      - Set different thresholds for each bus. Active buses use the same threshold and hysteresis values.
3. In the "Logic: Advanced" dialog, set the "Hysteresis" for each threshold to avoid the change of signal states due to noise.

14.2 Display

You can adjust the display of the parallel bus signals and the individual digital channels to optimize the analysis of bus data:

- Show the digital channels which are assigned to the bus, drag them to the optimal position, and scale them
- Show the decoded bus signal in different ways:
  – comb display with numeric bus values
  – analog display with bus values as amplitudes (quasi-analog waveform)

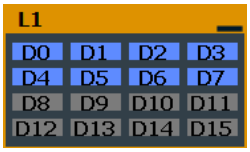
You can also drag the bus waveforms on the display and scale them.

- Show the result box of the decoded clocked bus signal

Each parallel bus is shown in a separate diagram, and the diagrams can be minimized and arranged as usual.

The signal icon indicates the activities on the digital channels even if they are not displayed in the diagram, of if the acquisition has been stopped:

- Blue: channel is low
- Green: channel is high
- Gray: channel state is changing



The display update rate of the oscilloscope is adapted to the visual perception of human eyes, and it is slower than the acquisition rate. All analog and digital waveforms that are acquired during one display update cycle are overlapped and displayed at once. Thus you can see the cumulative occurrence of binary states and edge transitions on the screen at once. Bus signals are not overlapped.

The trigger point is always visible on the display, it cannot be moved outside ("Restrict horizontal position to acquisition range" is enabled automatically).

If digital channels are active, the trigger point is always visible on the display, it cannot be moved outside.

To access and analyze one or more specific acquisitions, you can use the History Viewer in the common way.

Furthermore, you can zoom in digital signals and bus signal in the same way as in analog waveforms.

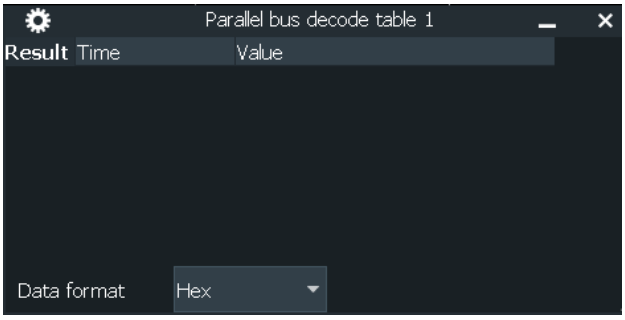
See also:

- Chapter 7.1, "Zoom", on page 239

14.2.1 Parallel bus - decode table

Decoding is available for clocked parallel buses.

The decode table shows the decoded data words of the bus signal and the corresponding time. Each clock edge corresponds to one row in the table. Below the table, you can select the data format of the bus values.



The results can be saved to a `.csv` or `.html` file, see Chapter 12.2.4, "Results", on page 467.

14.2.2 Adjusting the display of digital channels and parallel buses

The display of digital channels and parallel buses is flexible, you can adjust it to your needs by combining the following settings:

1. Enable "Show bus" if you want to display the bus signal in the diagram. Under "Bus representation", select if you want to display the decoded bus signal with bus values ("Comb"), or show the bus values as amplitudes, similar to an analog waveform ("Analog").
2. Check the signal icon of the bus to monitor the activities on the digital channels even if they are not displayed in the diagram:
   - Blue: channel is low
   - Green: channel is high
   - Gray: channel state is changing
3. In the diagram, you can change the display order of the digital channels by dragging the individual channels to the required position.
4. To adjust the line height and vertical position of all digital channels at once, tap one of the digital channels and turn the vertical [Scale] and [Position] rotary knobs. In the same way, you can move and scale the bus signal.
5. If the bus signal is displayed as quasi-analog waveform, you can double-tap the waveform to open the "Parallel buses" dialog box.
6. To switch off the display of the digital channels, disable "Show signals".

14.3 Trigger

For digital trigger sources are all trigger types useful that require only one trigger level as trigger condition. This level is the logical threshold. Possible trigger sources are the individual digital channels, parallel bus signals, or any logical combination of digital channels. The following trigger types are available:

Table 14-1: Trigger types and digital trigger sources

Trigger type	Trigger source is		
	Digital channel	Logic combination of digital channels	Parallel bus
Edge	X	X	
Width	X	X	
Timeout	X	X	
Setup & Hold	X		X
State		X	X
Pattern (with holdoff)		X	X
Serial Pattern	X	X	

For details, see: Chapter 14.3.1, "Trigger settings for digital signals and parallel buses", on page 997.

Also, you can define trigger holdoff conditions. See Chapter 6.4, "Mode / Holdoff", on page 220.

14.3.1 Trigger settings for digital signals and parallel buses

Depending on the selected source, the instrument provides the appropriate trigger types and the corresponding trigger settings.

The settings in the "Setup" tab are:

- Basic trigger settings....997
- Edge....998
- Width....999
- Timeout....1001
- Setup & Hold....1002
- State....1004
- Pattern....1005
- Serial pattern....1007

14.3.1.1 Basic trigger settings

The basic trigger settings for MSO are the trigger source and the trigger type. They are selected in the upper part of the "Trigger" dialog box.

Make sure that the trigger sequence is set to "A only".

Also, you can define trigger holdoff conditions. See Chapter 6.4, "Mode / Holdoff", on page 220.

Source
If the Mixed Signal Option is installed, the variety of trigger sources is enhanced with specific digital trigger sources. As a trigger source you can select:

- One of the digital channels "D0" ... "D15"

- A logic combination of digital channels: "Logic"

- One of the parallel buses "Par. bus1" ... "Par. bus4"

Remote command:
`TRIGger<m>:SOURce[:SELect]` on page 1389

Type
Depending on the selected source, the instrument provides the appropriate trigger types and the corresponding trigger settings. For mixed signal analysis, the following trigger types are available:

- Edge, see page 998
- Width, see page 999
- Timeout, see page 1001
- Setup & Hold, see page 1002
- State, see page 1004
- Pattern, see page 1005
- Serial pattern, see page 1007

Remote command:
`TRIGger<m>:PARallel:TYPE` on page 2432

14.3.1.2 Edge

Using the edge trigger, you can also trigger on a single digital channel (a logic bit), and a logical combination of digital channels.

Depending on the selected trigger source, different trigger settings are available. The trigger level is already set - in MSO the logical threshold is used as trigger level.

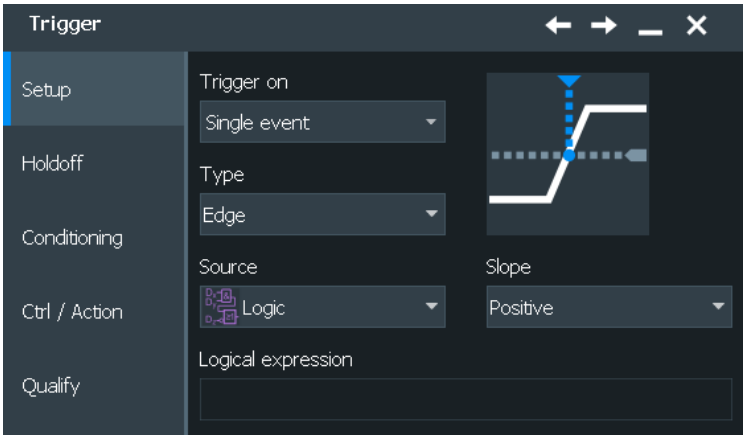


Figure 14-1: Edge trigger settings for trigger source = logical combination of digital channels (Logic)

Slope
Defines the edge - the state transition - of the signal.

"Positive" Means a 0 to 1 transition of the state.

"Negative" Means a 1 to 0 transition of the state.

"Either" Triggers on any activity on the selected trigger source.

Remote command:
`TRIGger<m>:PARallel:EDGE:SLOPe` on page 2433

Logical expression
Defines a logical combination of several digital channels as trigger condition if "Logic" is set for "Source".

If the "Slope" is positive, the trigger occurs when the logical expression becomes true. If the "Slope" is negative, the trigger occurs when the logical expression becomes false.

Remote command:
`TRIGger<m>:PARallel:EDGE:EXPRession[:DEFine]` on page 2433

14.3.1.3 Width

The width trigger detects positive and/or negative pulses of a pulse width (duration) inside or outside of a defined time limit. It can trigger on a single digital channel or a logical combination of digital channels.

The instrument triggers at the end of the detected pulse.

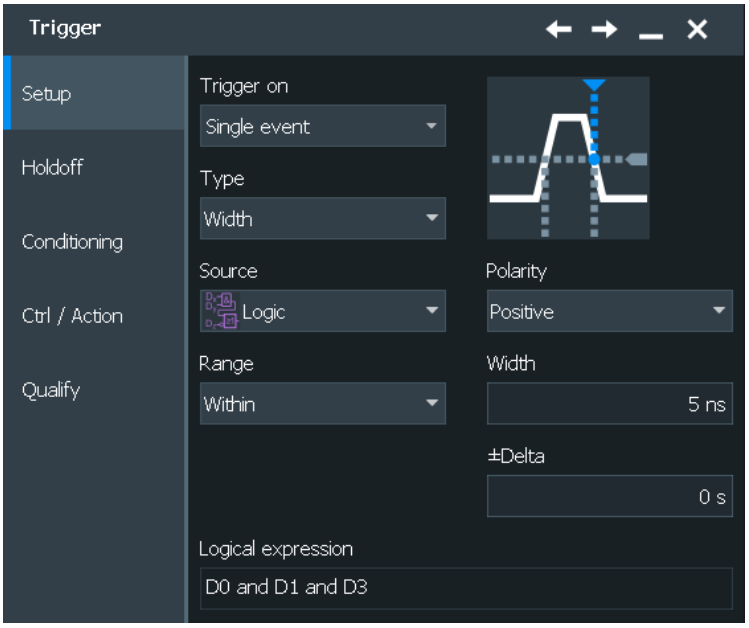


Figure 14-2: Width trigger settings for trigger source = logical combination of digital channels

Polarity

Sets the polarity of a pulse to "Positive", "Negative", or "Both".

When triggering on a positive pulse, the trigger event occurs on the high to low transition of the pulse if the timing condition is true. When triggering on a negative pulse, the trigger event occurs on the low to high transition of the pulse if the timing condition is true.

Remote command:

`TRIGger<m>:PARallel:WIDTh:POLarity` on page 2434

Logical expression

Defines a logical combination of several digital channels as trigger condition if "Logic" is set for "Source". As long as the digital signals match the logical expression (true), the pulse is positive. Otherwise, the pulse is negative.

Remote command:

`TRIGger<m>:PARallel:WIDTh:EXPRession[:DEFine]` on page 2433

Range

Selects how the range of a pulse width is defined:

"Within" Triggers on pulses inside a given time range. The time limit is defined by *Width ± Delta*.

"Outside" Triggers on pulses shorter or longer than a given time range. The time limit definition is the same as for "Within" range.

"Shorter" Triggers on pulses shorter than the given "Width".

"Longer" Triggers on pulses longer than the given "Width".

Remote command:

`TRIGger<m>:PARallel:WIDTh:RANGe` on page 2434

Width
For the ranges "Shorter" and "Longer", the width defines the maximum and minimum pulse width, respectively.

For the ranges "Within" and "Outside", the width defines the center of a range which is defined by the limits "±Delta".

Remote command:
`TRIGger<m>:PARallel:WIDTh:WIDTh` on page 2434

±Delta
Defines a range around the given width value.

The combination "Range" = Within and "±Delta" = 0 triggers on pulses with a pulse width that equals "Width".

The combination "Range" = Outside and "±Delta" = 0 means to trigger on pulse widths ≠ "Width".

Remote command:
`TRIGger<m>:PARallel:WIDTh:DELTa` on page 2435

14.3.1.4 Timeout

The timeout trigger event checks if the trigger source signal stays above or below the threshold voltage for a specified time lapse. In other words, the event occurs if the state condition remains unchanged for the specified time.

You can use the timeout trigger on a single digital channel, or a logical combination of digital channels.

Figure 14-3: Timeout trigger settings for trigger source = logical combination of digital channels

Range
Sets the state condition:

"Stays high" The level of a digital channel stays above the threshold, or the logical expression for "Logic" trigger source is true.

"Stays low" The level of a digital channel stays below the threshold, or the logical expression for "Logic" trigger source is false.

"High or low" The signal state remains unchanged.

Remote command:
`TRIGger<m>:PARallel:TIMeout:RANGe` on page 2435

Time
Defines the time limit for the timeout at which the instrument triggers.

Remote command:
`TRIGger<m>:PARallel:TIMeout:TIME` on page 2436

Logical expression
Defines a logic combination of several digital channels as trigger condition if "Logic" is set for "Source". The "Qualification Editor" supports the entry of the expression.

Remote command:
`TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine]` on page 2433
`TRIGger<m>:PARallel:STATe:EXPRession[:DEFine]` on page 2433
`TRIGger<m>:PARallel:PATTern:EXPRession[:DEFine]` on page 2433
`TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine]` on page 2433

14.3.1.5 Setup & Hold

The Setup & Hold trigger event occurs when the state of the trigger source signal changes inside a given time before the clock edge (setup time) or after the clock edge (hold time). The trigger event occurs at the clock edge for which the setup and/or hold time was violated.

With Setup & Hold trigger, you can trigger on a single digital channel, or a parallel bus to check several digital channels simultaneously. The clock signal is connected to one of the digital channels.

If you configure this trigger type for a parallel bus, the bus configuration is adjusted by the instrument if necessary. The bus is defined as clocked bus, and the clock source of the trigger is set as clock source of the bus.

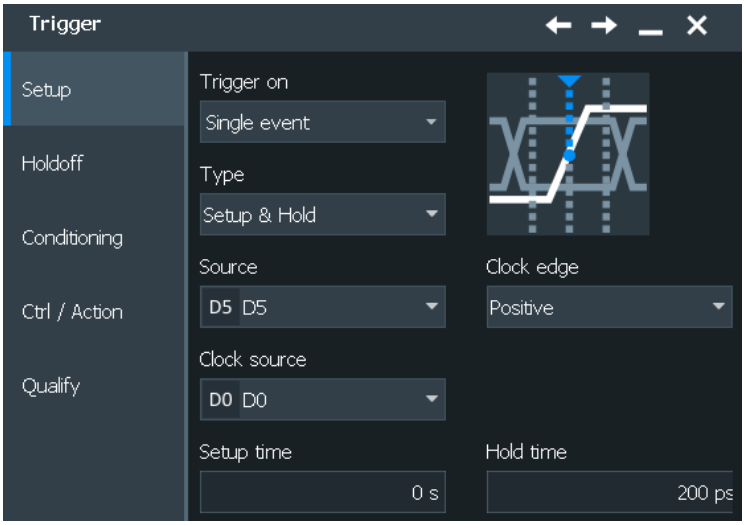


Figure 14-4: Setup & Hold trigger settings

Clock source
Selects the digital channel of the clock signal.

Remote command:
`TRIGger<m>:PARallel:SETHold:CSOurce[:VALue]` on page 2433
`TRIGger<m>:PARallel:STATe:CSOurce:VALue` on page 2433
`TRIGger<m>:PARallel:SPATtern:CSOurce[:VALue]` on page 2433

Clock edge
Sets the edge of the clock signal. The crossing of the clock edge and the logical threshold defines the time reference point for the setup and hold time measurement.

Remote command:
`TRIGger<m>:PARallel:SETHold:CSOurce:EDGE` on page 2436

Setup time
Sets the minimum time *before* the clock edge while data is stable and not change its state.

The setup time can be negative. In this case, the setup interval starts after the clock edge, and the hold time starts after the setup time has expired. Thus, the hold time is always positive. If you change the negative setup time, the hold time is adjusted by the instrument.

Remote command:
`TRIGger<m>:PARallel:SETHold:STIMe` on page 2437

Hold time
Sets the minimum time *after* the clock edge while data is stable and not change its state.

The hold time can be negative. In this case, the hold time ends before the clock edge, and the setup interval ends when the hold interval starts. Thus, the setup time is always positive. If you change the negative hold time, the setup time is adjusted by the instrument.

Remote command:
`TRIGger<m>:PARallel:SETHold:HTIMe` on page 2436

14.3.1.6 State

The state trigger detects the logical state of several logically combined digital channels at a given clock edge. The trigger source is a logical combination of digital channels or a parallel bus. The trigger occurs at the clock edge at which the state condition is true.

Figure 14-5: State trigger settings for trigger source = parallel bus

Clock source
Selects the digital channel of the clock signal.

Remote command:
`TRIGger<m>:PARallel:SETHold:CSOurce[:VALue]` on page 2433
`TRIGger<m>:PARallel:STATe:CSOurce:VALue` on page 2433
`TRIGger<m>:PARallel:SPATtern:CSOurce[:VALue]` on page 2433

Clock edge
Sets the edge of the clock signal. The crossing of the clock edge and the logical threshold defines the time at which the logical states and the bus value are analyzed.

Remote command:
`TRIGger<m>:PARallel:STATe:CSOurce:EDGE` on page 2437

Channel states
Available for "Trigger Source"= "Par. Bus1/2/3/4"

For each digital channel that is used in the bus, set the required state: 1, 0, or X (don't care).

Remote command:
`TRIGger<m>:PARallel:STATe:BIT<0..15>` on page 2437

Logical expression
Defines a logic combination of several digital channels as trigger condition if "Logic" is set for "Source". The "Qualification Editor" supports the entry of the expression.

Remote command:
`TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine]` on page 2433
`TRIGger<m>:PARallel:STATe:EXPRession[:DEFine]` on page 2433
`TRIGger<m>:PARallel:PATTern:EXPRession[:DEFine]` on page 2433
`TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine]` on page 2433

14.3.1.7 Pattern

The pattern trigger identifies a logical state of several logically combined digital channels (pattern) and a time limitation (holdoff). The pattern definition is defined by the logical expression, if "Logic" is used for trigger source. For a parallel bus trigger source, the pattern is defined by setting the state of each digital channel.

The timing starts when the pattern comes true. The decision level is the logical threshold.

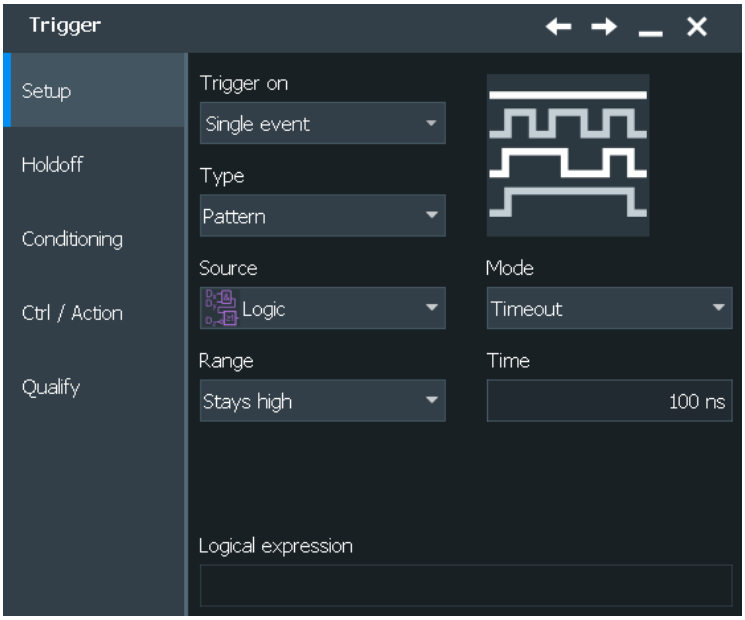


Figure 14-6: Pattern trigger settings for trigger source = parallel bus and timeout

Channel states
Available for "Trigger Source"= "Par. Bus1/2/3/4"

For each digital channel that is used in the bus, set the required state: 1, 0, or X (do not care).

Remote command:
`TRIGger<m>:PARallel:PATTern:BIT<0..15>` on page 2437

Logical expression
Defines a logic combination of several digital channels as trigger condition if "Logic" is set for "Source". The "Qualification Editor" supports the entry of the expression.

Remote command:
`TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine]` on page 2433
`TRIGger<m>:PARallel:STATe:EXPRession[:DEFine]` on page 2433
`TRIGger<m>:PARallel:PATTern:EXPRession[:DEFine]` on page 2433
`TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine]` on page 2433

Mode: Off, Timeout, Width
Sets the mode of the timing condition.

"Off" No timing condition, only the logical pattern condition is relevant.

"Timeout" Defines a minimum time qualification to avoid triggering on unstable or transitional conditions. Even in best-designed systems, there are slight delays between the signal when digital signals change states. This means that there are always transitional state conditions when signals are switching.
See "Timeout settings" on page 1006 for a description of the settings.
The trigger event occurs when the pattern stays unchanged for the specified time.

"Width" Sets a pulse width as timing condition, see "Width settings" on page 1007. The pulse starts when the pattern comes true, and the trigger event occurs when the pattern comes false during the specified time limit.
Using this mode, you can, for example, trigger exclusively on unstable conditions - if the pattern is present for less than a specified time.

Remote command:
`TRIGger<m>:PARallel:PATTern:MODE` on page 2438

Timeout settings
The timeout settings "Range" and "Time" appear if the timing mode is set to "Timeout".

Range ← Timeout settings
Sets the state condition:

"Stays high" The pattern stays true for the specified time.

"Stays low" The pattern stays false for the specified time.

"High or low" The pattern remains unchanged for the specified time.

Remote command:
`TRIGger<m>:PARallel:PATTern:TIMeout:MODE` on page 2438

Time ← Timeout settings
Defines the time limit for the timeout at which the instrument triggers.

Remote command:
`TRIGger<m>:PARallel:PATTern:TIMeout[:TIME]` on page 2438

Width settings
The width settings "Range", "Width" and "±Delta" appear if the timing mode is set to "Width".

Range ← Width settings
Selects how the range of a pulse width is defined:

"Within"	Triggers when the pattern comes false inside a given time range. The time limit is defined by *Width ± Delta*.
"Outside"	Triggers when the pattern comes false before or after the given time range. The time limit definition is the same as for "Within" range.
"Shorter"	Triggers when the pattern comes false before the given "Width" has expired.
"Longer"	Triggers when the pattern comes false after the given "Width" has expired.

Remote command:
`TRIGger<m>:PARallel:PATTern:WIDTh:RANGe` on page 2439

Width ← Width settings
For the ranges "Shorter" and "Longer", the width defines the maximum and minimum time limit, respectively.

For the ranges "Within" and "Outside", the width defines the center of a range which is defined by the limits "±Delta".

Remote command:
`TRIGger<m>:PARallel:PATTern:WIDTh[:WIDTh]` on page 2439

±Delta ← Width settings
Defines a range around the given width value.

The combination "Range" = Within and "±Delta" = 0 triggers on pulses with a pulse width that equals "Width".

The combination "Range" = Outside and "±Delta" = 0 means to trigger on pulse widths ≠ "Width".

Remote command:
`TRIGger<m>:PARallel:PATTern:WIDTh:DELTa` on page 2440

14.3.1.8 Serial pattern

The serial pattern trigger identifies a serial bit string trigger on a single digital channel, or for a logical combination of digital channels. The trigger requires a clocked bus; the bits are read at the specified clock edge. The trigger event occurs at the last clock edge of the serial bit string.

This trigger type allows you to trigger on specific address or data transmissions in serial input and output signals.

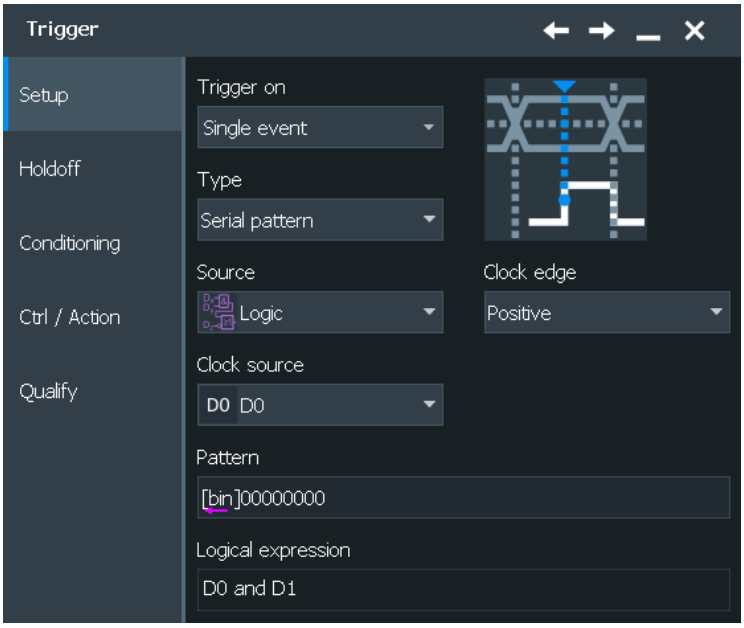


Figure 14-7: Serial pattern trigger settings for trigger source = logical combination of digital channels

Clock edge
Sets the edge of the clock signal. The bit value is determined at the crossing of the clock edge and the logical threshold.

Remote command:
`TRIGger<m>:PARallel:SPATtern:CSOurce:EDGE` on page 2440

Clock source
Selects the digital channel of the clock signal.

Remote command:
`TRIGger<m>:PARallel:SETHold:CSOurce[:VALue]` on page 2433
`TRIGger<m>:PARallel:STATe:CSOurce:VALue` on page 2433
`TRIGger<m>:PARallel:SPATtern:CSOurce[:VALue]` on page 2433

Logical expression
Defines a logic combination of several digital channels as trigger condition if "Logic" is set for "Source". The "Qualification Editor" supports the entry of the expression.

Remote command:
`TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine]` on page 2433
`TRIGger<m>:PARallel:STATe:EXPRession[:DEFine]` on page 2433
`TRIGger<m>:PARallel:PATTern:EXPRession[:DEFine]` on page 2433
`TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine]` on page 2433

Pattern
Defines the serial bit string on which to trigger. Touch and hold the "Pattern" field to open the "Bit Pattern Editor" where you can enter the pattern in various formats. The pattern has to be defined exactly, X (do not care) is not supported in binary format.

See also: Chapter 13.1.7, "Bit pattern editor", on page 493

Remote command:
`TRIGger<m>:PARallel:SPATtern:PATTern` on page 2440

14.3.2 Triggering on digital signals and parallel buses

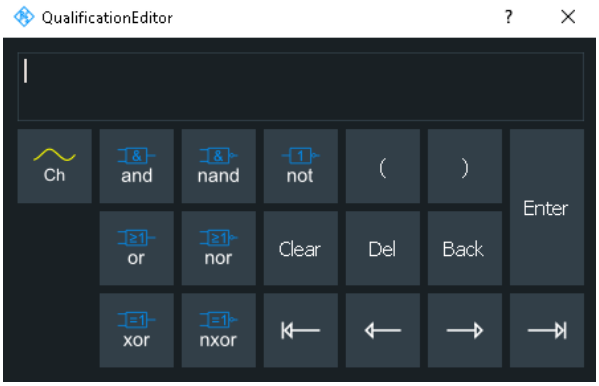
For a detailed description of the settings, see Chapter 14.3.1, "Trigger settings for digital signals and parallel buses", on page 997.

1. Open "Menu">"Trigger" > "Setup" tab
2. Select the trigger "Source":
   - One of the digital channels "D0" ... "D15"

   - A logic combination of digital channels: "Logic"

   - One of the parallel buses "Par. bus1" ... "Par. bus4"

3. Select the trigger "Type".
4. Under "Trigger type dependent settings", configure the trigger.
5. For trigger source "Logic", enter the logical expression of the digital channel combination. Tap and hold the "Logical expression" field until the "Qualification Editor" opens. It provides all logic operators that can be used in the expression.



14.4 Measurements on digital channels

For measurements on digital channels, the number of measurement categories and types is reduced to applicable measurement.

These are:

- Period
- Frequency
- Positive and negative pulse
- Pulse count
- Delay
- Phase
- Positive and negative duty cycle
- Burst width
- Edge count

Except for delay measurement, the measurements have the same settings and results for analog and digital sources.

Delay measurement on digital channels is reduced to measure the time between two subsequent rising or two subsequent falling edges.

See also Chapter 8.2, "Automatic measurements", on page 295.

14.5 Data export

The data of digital channels and parallel buses can be saved to file in the same way as analog waveform data. One digital channel or bus per file can be saved.

The data format of the stored values is defined with Menu > "Logic" > "Bus" tab > "Data format".

If the data is written to XML or CSV files, the selected format is used. If the target file format is BIN, you can save signed and unsigned binary data. The data format "Signed" writes signed data; all other formats are saved as unsigned binary data.

The export of raw data is not available.

Export of a digital channel

If the data of digital channels is stored in BIN format, 1 bit is written for each sample. 8 data samples are written in 1 byte (data word). Thus, the file size is

File size = Number of samples / 8

For example, 100 MSa are written into a 12.5 MByte BIN file. After reading the file, you have to extract the samples from the data words.

If saved to BIN file, the digital channel can be imported as reference waveform.

Export of a parallel bus

A parallel bus can be exported to file if "Enable bus" and "Show bus" are both activated, and the bus representation is "Comb".

All data formats can be saved to XML, CSV, and BIN files. If you save binary format to XML or CSV, you can see the values of each line for each sample.

In BIN files, 4 Bytes are written for each sample.

See also:

- Chapter 12.2.7, "Saving and loading waveform data", on page 473
- Chapter 12.2.1, "Waveform export files", on page 453

Remote commands for export to file:

- `EXPort:WAVeform:SOURce` on page 1649
- `EXPort:WAVeform:NAME` on page 1651
- `EXPort:WAVeform:SAVE` on page 1651

Remote commands for remote data transfer:

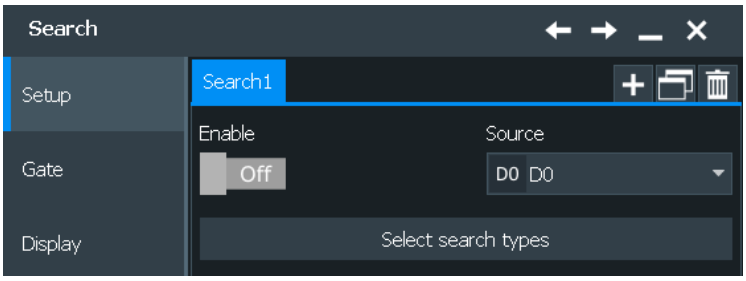
- `BUS<m>:PARallel:DATA:FORMat` on page 2427
- `BUS<m>:PARallel:DATA:HEADer?` on page 2442
- `BUS<m>:PARallel:DATA[:VALues]?` on page 2442
- `DIGital<m>:DATA:HEADer?` on page 2441
- `DIGital<m>:DATA[:VALues]?` on page 2441

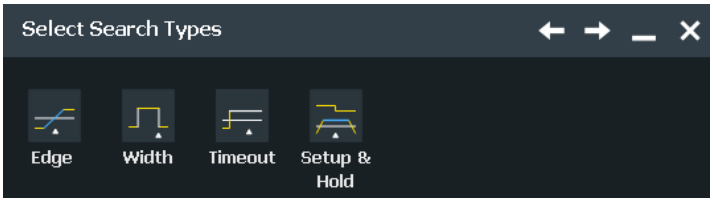
14.6 Mathematics

A parallel bus that is displayed as quasi-analog waveform can be analyzed with FFT. To configure the FFT, use the "Advanced" mode and the formula editor.

14.7 Search

Access: "Menu" > "Apps" > "Analysis" tab > "Search".





It is also possible to search on digital channels for specified events. Search conditions use the same parameters as the trigger event definition, see Chapter 14.3.1, "Trigger settings for digital signals and parallel buses", on page 997.

You can search for edge, width, timeout, and Setup & Hold conditions.

15 Waveform generator (Option R&S RTO6-B6)

The R&S RTO6 includes an integrated waveform generator which can generate input signals and patterns during testing. You can setup and output two waveform generators and a pattern generator.

With each of the waveform generators, one can output simple functions, modulated sine waveform, arbitrary waveforms and sweep waveforms. It is possible to couple and synchronize the settings of the waveform generators.

The instrument preset does not affect the generator settings. Each generator has its own preset.

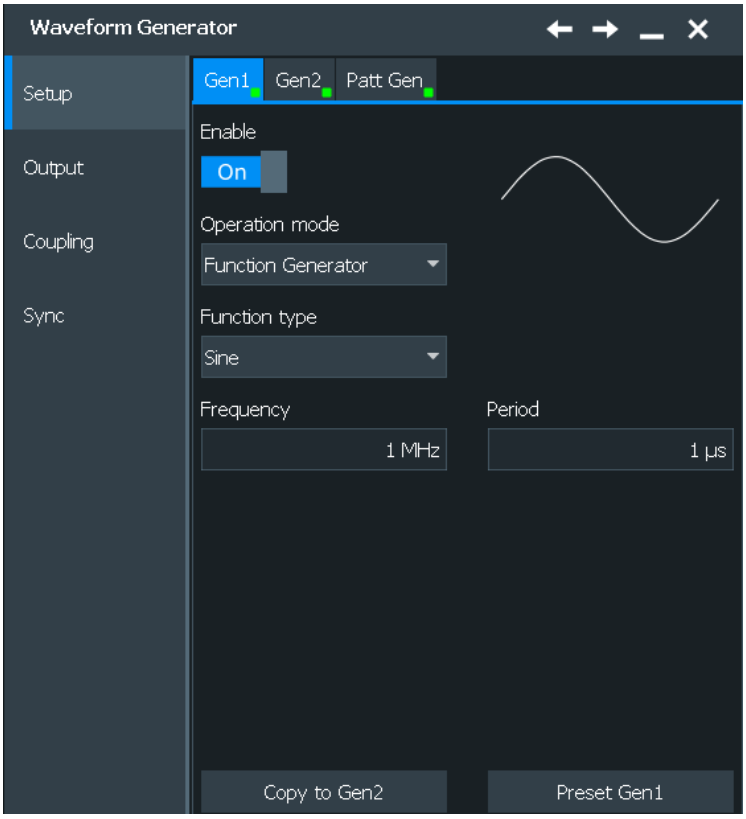
Always use double-shielded cables to prevent electromagentic interference (EMI) from impairing measurement results.

15.1 Setup of the waveform generator

Access: [Gen] > "Setup" tab

The "Setup" tab is divided into several sections:

- General settings like enabling and presetting the waveform generator
- Signal settings, depending on the selected Operation mode refer to one of the following:
  - Chapter 15.1.2, "Function generator", on page 1015
  - Chapter 15.1.3, "Modulation", on page 1020
  - Chapter 15.1.4, "Sweep", on page 1026
  - Chapter 15.1.5, "Arbitrary", on page 1027



Make sure that the tab of the correct waveform generator is selected on the left side.

The settings of the waveform generators are not affected by an instrument preset. Press "Preset Gen1/2" to preset the settings of the corresponding waveform generator.

15.1.1 General settings

The general waveform generator settings are for enabling and presetting the generator and selecting the "Operation Mode".

Enable
Enables the waveform generator/ pattern generator and outputs the signal to the connectors.

Remote command:
`WGENerator<m>[:ENABle]` on page 2443
`PGENerator:ENABle` on page 2456

Operation mode
Selects the operation mode for the waveform generator. The "Signal" settings depend on the selected mode.

For the settings of the different operation modes, refer to:

- Chapter 15.1.2, "Function generator", on page 1015
- Chapter 15.1.3, "Modulation", on page 1020
- Chapter 15.1.4, "Sweep", on page 1026
- Chapter 15.1.5, "Arbitrary", on page 1027

Remote command:
`WGENerator<m>:SOURce` on page 2443

Copy settings from Gen1/Gen2
Copies all settings from Gen1/Gen2 and applies them to Gen2/Gen1.

Remote command:
`WGENerator<m>:ACOPy` on page 2443

Preset Gen1/Gen2/Patt Gen
Sets the parameters of the generator to their default values. The settings of the generators are not affected by an instrument preset.They are also not stored in the user-defined preset.

Remote command:
`WGENerator<m>:PRESet` on page 2443
`PGENerator:PRESet` on page 2457

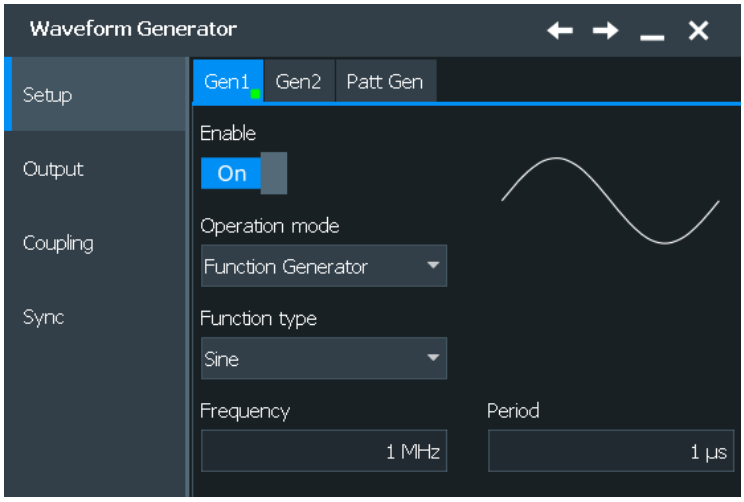
Message
Displays relevant messages concerning the coupling and sync settings.

15.1.2 Function generator

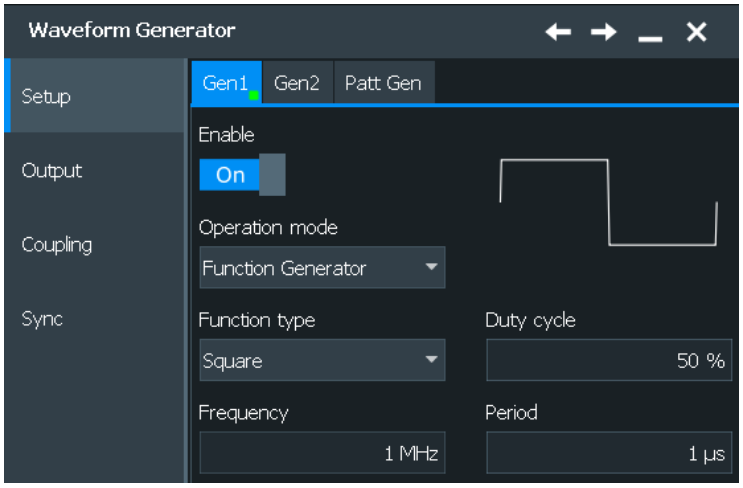
The R&S RTO6 function generator can generate input signals during testing. These signals can be used, for example, when testing circuits.

Function type
Selects the type of waveform to be generated for the function generator.

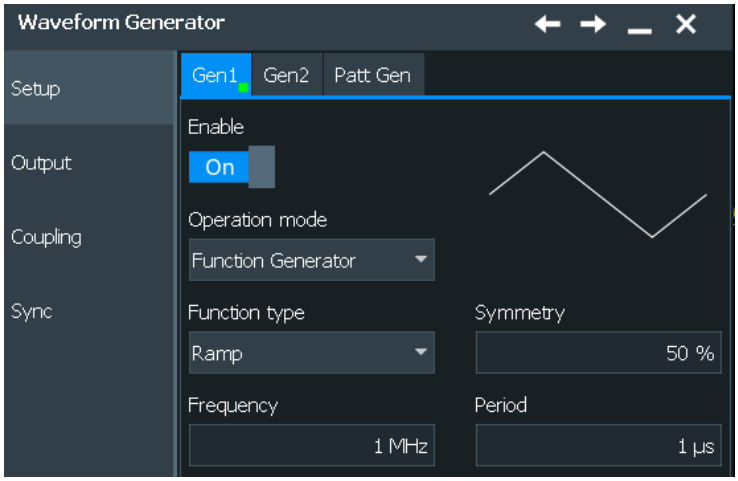
"Sine" Generates a sine wave. You can set its Frequency and Period.



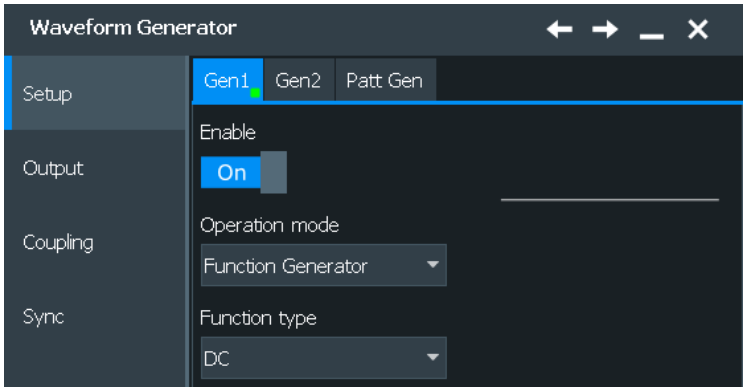
"Square" Generates a square wave. You can set its Frequency, Period and Duty cycle.



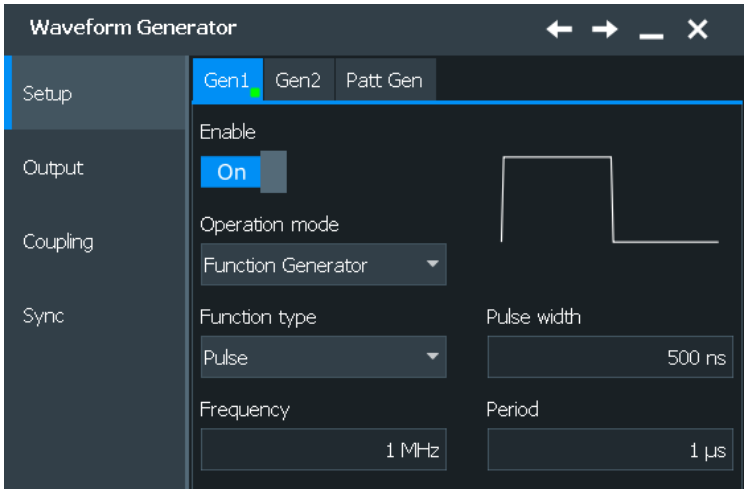
"Ramp" Generates a ramp signal. You can set its Frequency, Period and Symmetry.



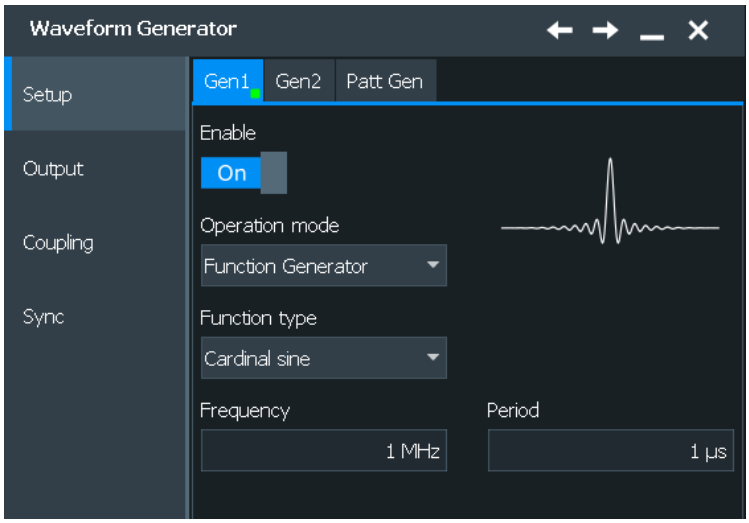
"DC" Generates a direct current (DC) signal. You can set the "Absolute level" on page 1037.



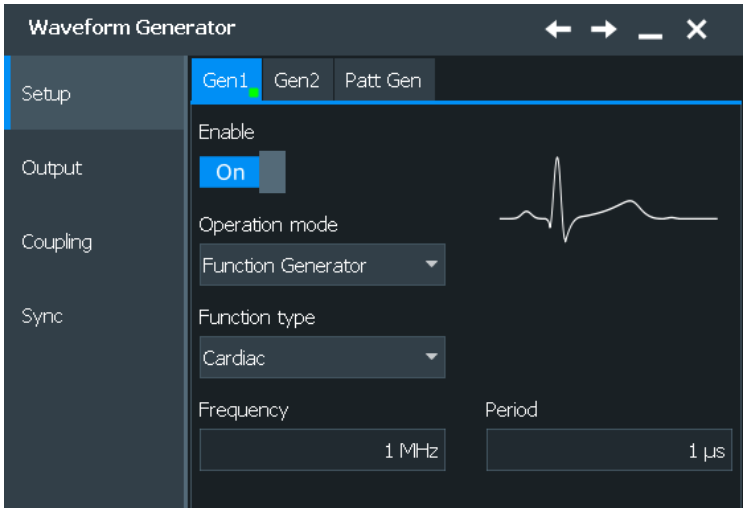
"Pulse" Generates a pulse signal. You can set the Frequency, Period and Pulse width.



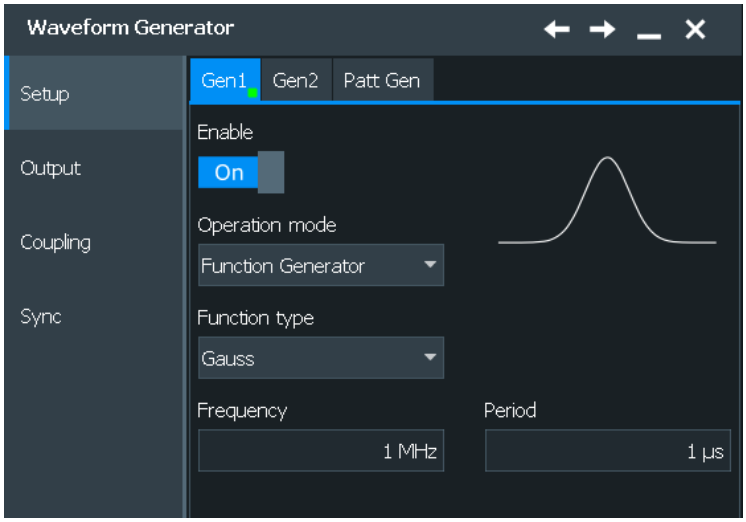
"Cardinal sine" Generates a cardinal sine wave. You can set the Frequency and Period.



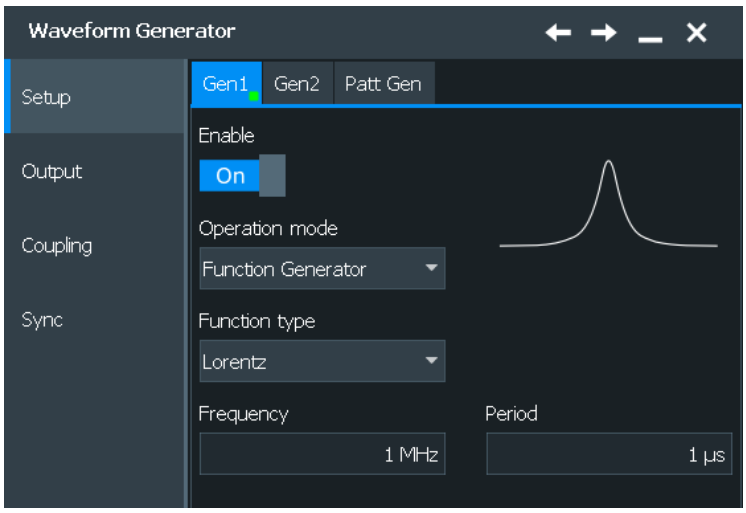
"Cardiac" Generates a cardiac signal. You can set the Frequency and Period.



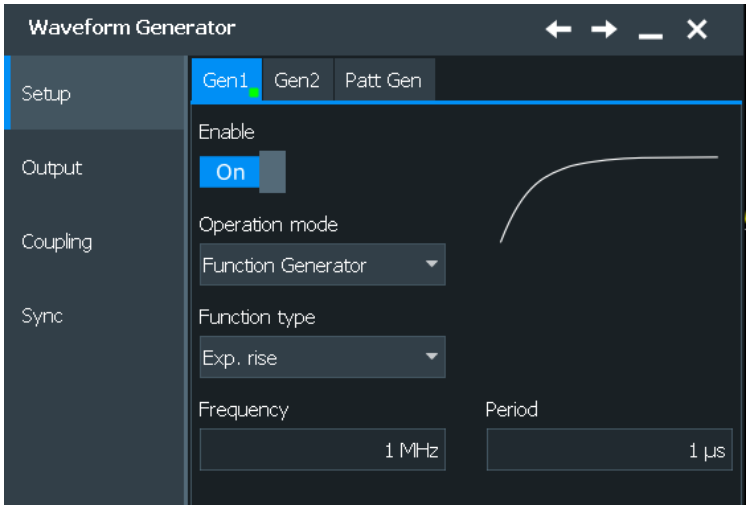
"Gauss" Generates a gauss signal . You can set the Frequency and Period.



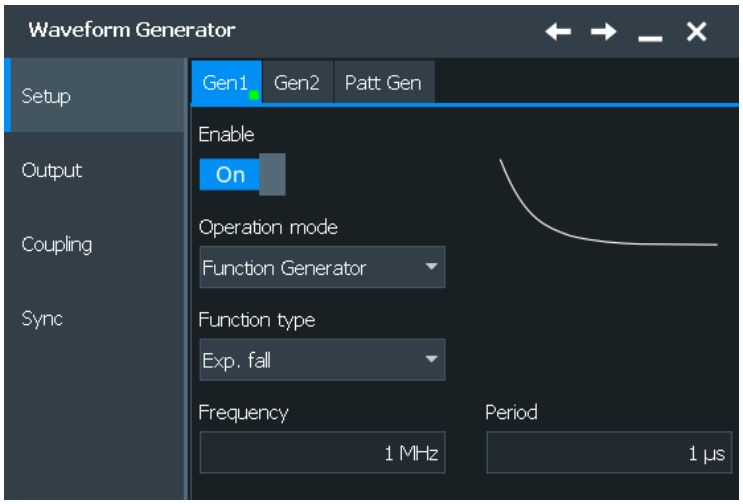
"Lorentz" Generates a Lorentz signal . You can set the Frequency and Period.



"Exp. rise" Generates an exponential rise signal. You can set the Frequency and Period.



"Exp. fall" Generates an exponential fall signal. You can set the Frequency and Period.



Remote command:
`WGENerator<m>:FUNCtion[:SELect]` on page 2444

Frequency
Sets the frequency of the waveform. The available frequency range depends on the selected "Function Type", see Frequency range of the function generator waveforms.

Table 15-1: Frequency range of the function generator waveforms

"Function type"	Min frequency	Max frequency
"Sine"	0.001 Hz	100 MHz
"Square"	0.001 Hz	30 MHz
"Ramp"	0.001 Hz	1 MHz
"DC"	-	-

"Function type"	Min frequency	Max frequency
"Pulse"	0.001 Hz	30 MHz
"Cardinal sine"	0.001 Hz	5 MHz
"Cardiac"	0.001 Hz	1 MHz
"Gauss"	0.001 Hz	25 MHz
"Lorentz"	0.001 Hz	10 MHz
"Exp.rise"	0.001 Hz	10 MHz
"Ep. fall"	0.001 Hz	10 MHz

The values of the "Frequency" and "Period" depend on each other, as:

Period = 1/ Frequency

Remote command:
`WGENerator<m>:FREQuency` on page 2444

Period
Sets the period of the waveform. The available period range depends on the selected "Function Type".

Remote command:
`WGENerator<m>:PERiod` on page 2444

Duty cycle
Sets the duty cycle for a square waveform. The duty cycle expresses for what percentage of the period, the signal state is high.

Remote command:
`WGENerator<m>:FUNCtion:SQUare:DCYCle` on page 2445

Symmetry
Sets the symmetry of a ramp waveform, the percentage of time the waveform is rising. By changing the symmetry of the ramp, you can create, for example, triangular waveforms.

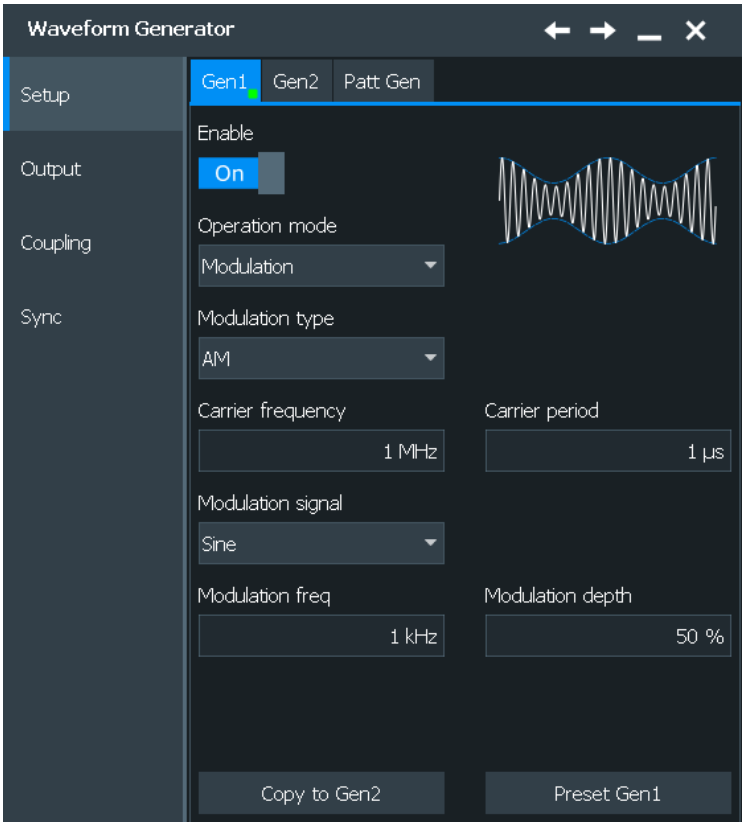
Remote command:
`WGENerator<m>:FUNCtion:RAMP[:SYMMetry]` on page 2445

Pulse width
Sets the pulse width, the pulse duration of the generated pulse waveform.

Remote command:
`WGENerator<m>:FUNCtion:PULSe[:WIDTh]` on page 2445

15.1.3 Modulation

Modulation is when properties of an original periodic waveform, the carrier signal, are varied according to a second modulating signal. The type of modulation used determines which properties are changed.



15.1.3.1 General settings

Consists of settings for selecting the modulation type.

Modulation type
Selects the modulation type, which defines how the carrier signal is modified.

"AM"	Amplitude modulation. See: Chapter 15.1.3.2, "AM modulation", on page 1022.
"FM"	Frequency modulation. See: Chapter 15.1.3.4, "FM modulation", on page 1024.
"PWM"	Pulse width modulation See: Chapter 15.1.3.3, "PWM modulation", on page 1023
"FSK"	Frequency shift keying (FSK) modulation. See:Chapter 15.1.3.5, "FSK modulation", on page 1026.

Remote command:
`WGENerator<m>:MODulation:TYPE` on page 2446

Carrier frequency
Sets the frequency of the carrier signal.

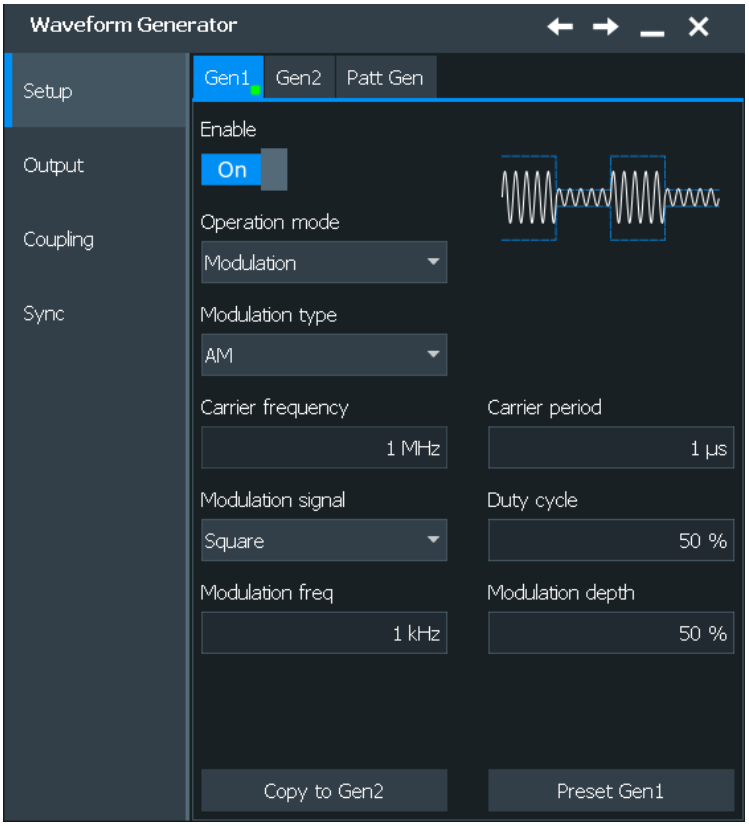
Remote command:
`WGENerator<m>:MODulation:CARRier:FREQuency` on page 2447

Carrier period
Sets the period of the carrier signal.

Remote command:
`WGENerator<m>:MODulation:CARRier:PERiod` on page 2447

15.1.3.2 AM modulation

For amplitude modulation (AM), the amplitude of the carrier signal is varied according to the modulation signal.



Modulation signal
Selects the type of the modulation signal for the AM modulation types.

Remote command:
`WGENerator<m>:MODulation:AM[:FUNCtion]` on page 2448

Modulation freq
Sets the frequency of the modulation waveform.

Remote command:
`WGENerator<m>:MODulation:AM:FREQuency` on page 2447

Modulation depth
Sets the modulation depth, the percentage of the amplitude range that is used for the modulation.

Remote command:
`WGENerator<m>:MODulation:AM:DEPTh` on page 2446

Symmetry
Sets the symmetry for the ramp modulation waveform, the percentage of time that the waveform is rising.

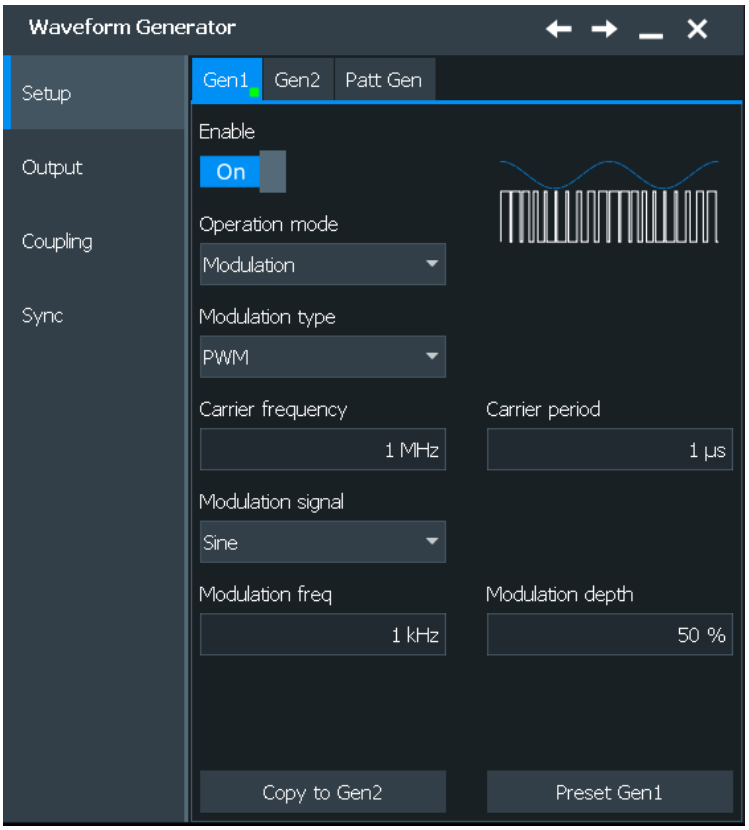
Remote command:
`WGENerator<m>:MODulation:AM:SYMMetry` on page 2447

Duty cycle
Sets the duty cycle for a square waveform. The duty cycle expresses for what percentage fraction of the period, the waveform is active, i.e. the signal state is high.

Remote command:
`WGENerator<m>:MODulation:AM:DCYCle` on page 2446

15.1.3.3 PWM modulation

For pulse width modulation (PWM), the time for which the signal is in a high state is varied according to the modulation signal.



Modulation signal
Selects the type of the modulation signal for the PWM modulation types.

Remote command:
`WGENerator<m>:MODulation:PWM[:FUNCtion]` on page 2451

Modulation freq
Sets the frequency of the modulation waveform.

Remote command:
`WGENerator<m>:MODulation:PWM:FREQuency` on page 2451

Modulation depth
Sets the modulation depth, the percentage of the pulse width range that is used for the modulation.

Remote command:
`WGENerator<m>:MODulation:PWM:DEPTh` on page 2450

Symmetry
Sets the symmetry for the ramp modulation waveform, the percentage of time that the waveform is rising.

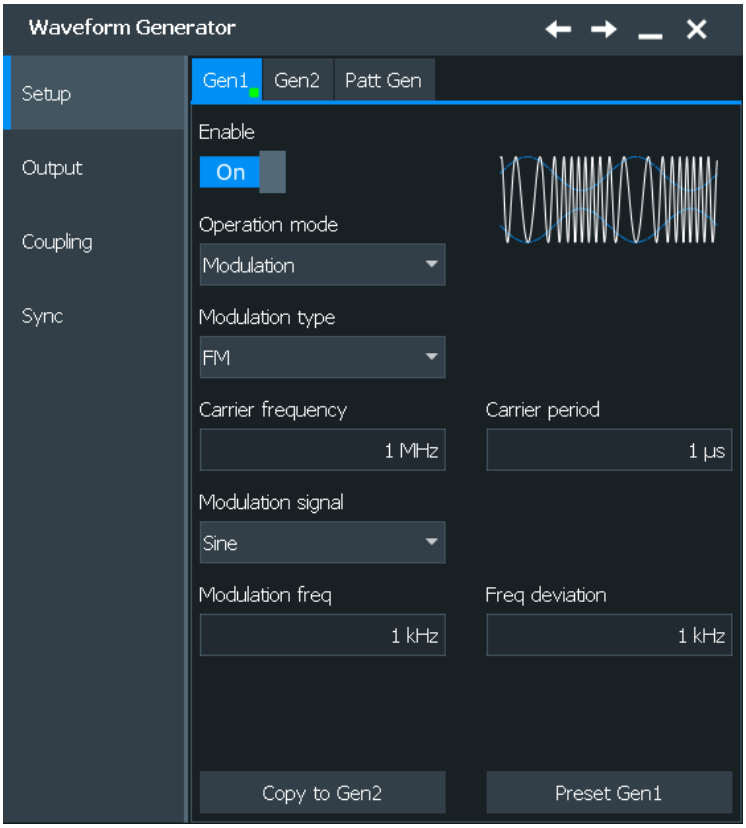
Remote command:
`WGENerator<m>:MODulation:PWM:SYMMetry` on page 2451

Duty cycle
Sets the duty cycle for a square waveform. The duty cycle expresses for what percentage fraction of the period, the waveform is active, i.e. the signal state is high.

Remote command:
`WGENerator<m>:MODulation:PWM:DCYCle` on page 2450

15.1.3.4 FM modulation

For frequency modulation (FM), the frequency of the carrier signal is varied according to the modulation signal.



Modulation signal
Selects the type of the modulation signal for the FM modulation types.

Remote command:
`WGENerator<m>:MODulation:FM[:FUNCtion]` on page 2449

Modulation freq
Sets the frequency of the modulation waveform.

Remote command:
`WGENerator<m>:MODulation:FM:FREQuency` on page 2448

Symmetry
Sets the symmetry for the ramp modulation waveform, the percentage of time that the waveform is rising.

Remote command:
`WGENerator<m>:MODulation:FM:SYMMetry` on page 2449

Freq deviation
Sets the frequency deviation, the maximum difference between and FM modulated signal and the carrier signal.

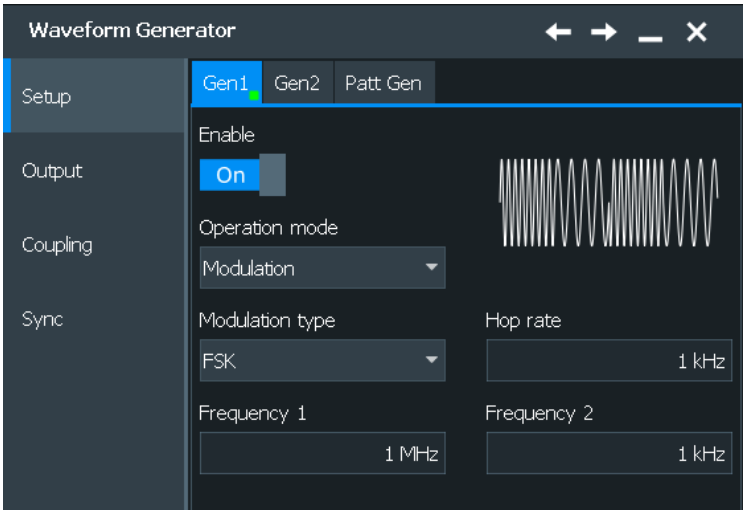
Remote command:
`WGENerator<m>:MODulation:FM:DEViation` on page 2448

Duty cycle
Sets the duty cycle for a square waveform. The duty cycle expresses the percentage of the period during which the waveform is active, i.e. the signal state is high.

Remote command:
`WGENerator<m>:MODulation:FM:DCYCle` on page 2448

15.1.3.5 FSK modulation

For frequency shift keying (FSK) modulation, the signal switches between Frequency 1 and Frequency 2 at a Hop rate.



Frequency 1
Sets the frequency of the first signal in FSK modulated signal.

Remote command:
`WGENerator<m>:MODulation:FSK:FONE` on page 2449

Frequency 2
Sets the frequency of the second signal in FSK modulated signal.

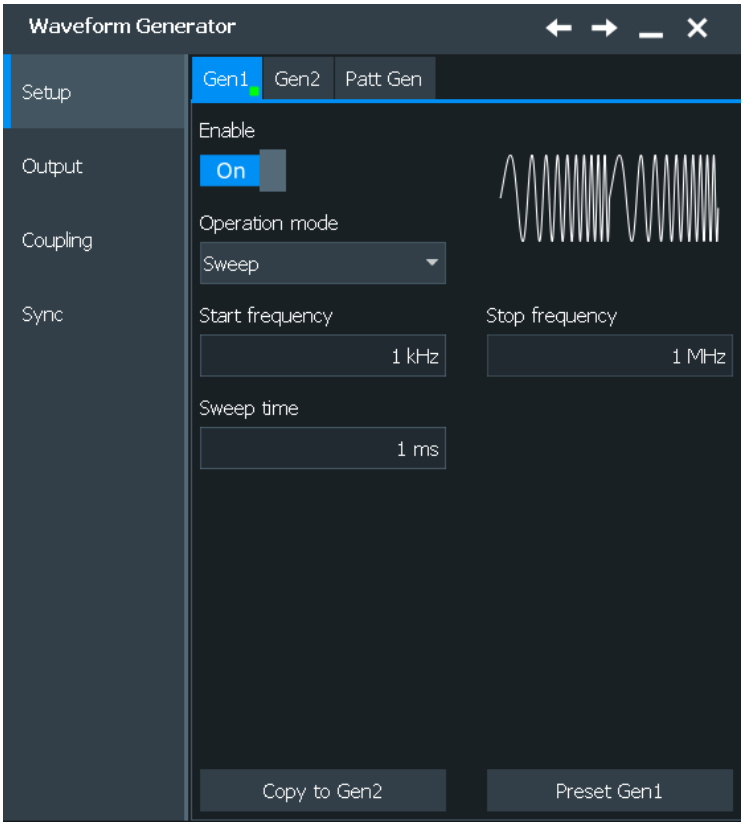
Remote command:
`WGENerator<m>:MODulation:FSK:FTWO` on page 2450

Hop rate
Sets the hop rate, the frequency at which signal switches between Frequency 1 and Frequency 2.

Remote command:
`WGENerator<m>:MODulation:FSK[:RATE]` on page 2450

15.1.4 Sweep

In the sweep mode, the R&S RTO6 generates a signal whose frequency gradually changes from the Start frequency to the Stop freuqency for a certain Sweep time.



Start frequency
Sets the start frequency of the sweep signal.

Remote command:
`WGENerator<m>:SWEep:FSTart` on page 2452

Stop freuqency
Sets the stop frequency of the sweep signal.

Remote command:
`WGENerator<m>:SWEep[:FEND]` on page 2452

Sweep time
Sets the duration of the sweep.

Remote command:
`WGENerator<m>:SWEep:TIME` on page 2452

15.1.5 Arbitrary

The arbitrary waveform generator allows you to output a user-defined waveform for testing your devices. You can output a waveform from a file or from the current R&S RTO6 reference curve format. Files in `*.csv` and `*.bin` formats are supported. This files must follow a defined structure. You can load `*.csv` files in an R&S Wave

Gen format (see Content and format of the R&S wave gen *.csv files), Tektronix AFG format or Keysight WaveGen format.

Content and format of the R&S wave gen *.csv files

The R&S waveform generator format can contain the following values:

- Rate
- Time value
- Voltage value

If all the values are defined, the file format is as follows:

```
Rate = 5000000           //Sample rate of the arbitrary waveform.
0.000000E+000,-5.995    //Time value 1, Voltage value 1
1.237011E-005,-6.0      //Time value 2, Voltage value 2
.......
```

In this case, the rate is reflected in the "Sample Rate" field of the user interface. The total number of Time/Voltage values is reflected in the "Samples" of the user interface. Anything written after `//` is ignored as a comment.

You can define only some of the values. According to what you define, the file format looks different and is handled differently:

- With specified *Rate*:
  Time values are ignored. You can specify just rate and voltage values as below:

```
Rate = 5000000              //Sample rate of the arbitrary waveform
-5.995                      //Voltage value 1
-6.0                        //Voltage value 2
.......
```

- Without specified *Rate* and without specified *Time* values:
  The last user-defined sample rate is used to calculate the waveform. You can change the "Sample rate" in the user interface. The voltage values are then played with this sample rate.

```
-5.995                //Voltage value 1
-6.0                   //Voltage value 2
.......
```

- Without specified *Rate* and wit specified *Time*:
  The timing information of the first 2 time values is used to calculate the sample rate.

Example:

Consider the following file:

```
0.000000E+000,-5.995    //Time value 1, Voltage value 1
1.237011E-005,-6.0      //Time value 2, Voltage value 2
```

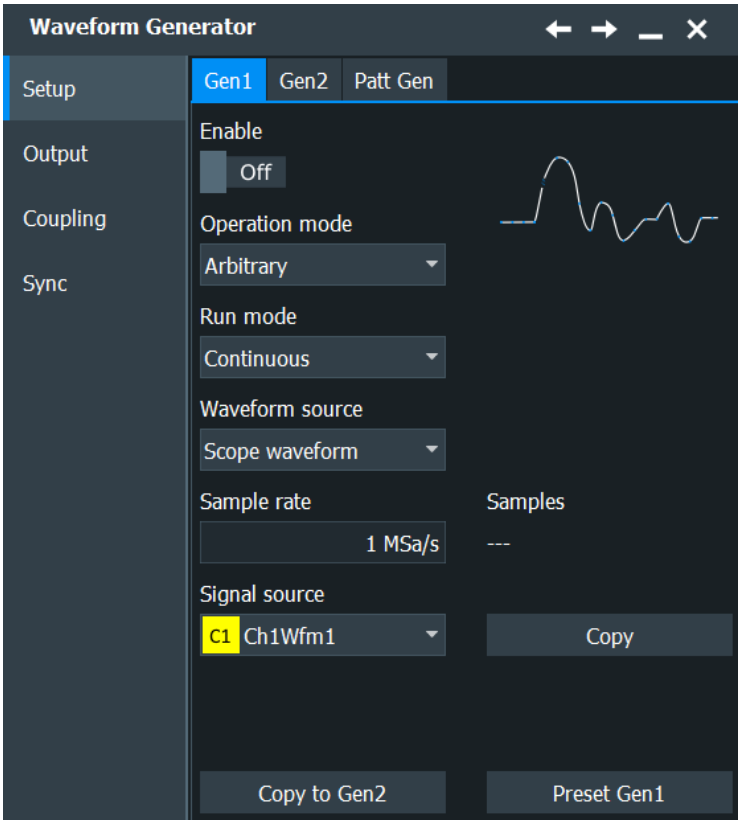
The sample rate is:

Sample rate = 1/ Time between first two samples =1 / 1.237011E-005 = 80.840KSampe/sec

Content and format of the R&S arbitrary generator *.bin files

The file stream should contain the following information in the given order:

- Sample rate [double format]
- Number of samples [unsigned integer format]
- Samples [double format] * number of samples



15.1.5.1 General settings

Waveform source
Selects the arbitrary waveform source. You can load an existing file or load the current oscilloscope waveform.

Remote command:
`WGENerator<m>:ARBGen[:SOURce]` on page 2455

Run mode
Selects the duration for which the signal of the arbitrary generator will be output after the trigger event. You can choose between a "Continuous" and "Single period" duration.

Remote command:
`WGENerator<m>:ARBGen:RUNMode` on page 2454

Sample rate
Sets the sample rate for the arbitrary waveform.

Remote command:
`WGENerator<m>:ARBGen:SRATe` on page 2455

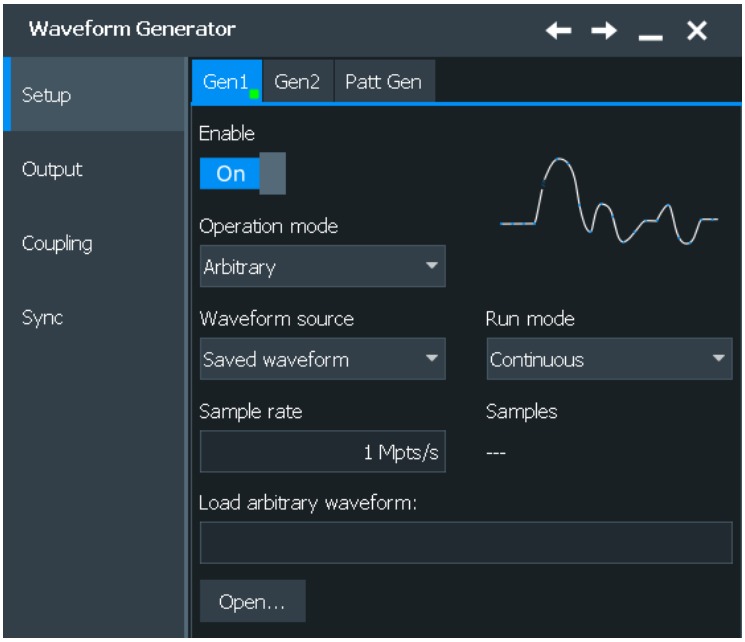
Samples
Displays the number of samples in the loaded waveform.

Remote command:
`WGENerator<m>:ARBGen:SAMPles?` on page 2454

Run single
Generates a single period of the arbitrary waveform if "Run mode" is set to "Single period".

Remote command:
`WGENerator<m>:ARBGen:RUNSingle` on page 2456

15.1.5.2 Saved waveform



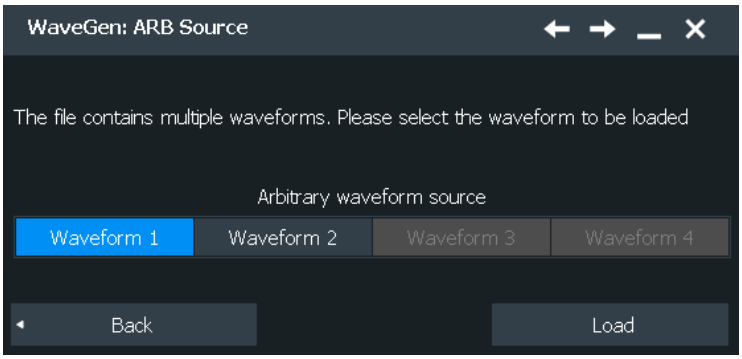
For a saved waveform, the following settings are available:

Load arbitrary waveform
Opens a file selection dialog box and loads the selected file. Supported are `.bin` and `.csv` extension files.

Remote command:
`WGENerator<m>:ARBGen:NAME` on page 2454
`WGENerator<m>:ARBGen:OPEN` on page 2454

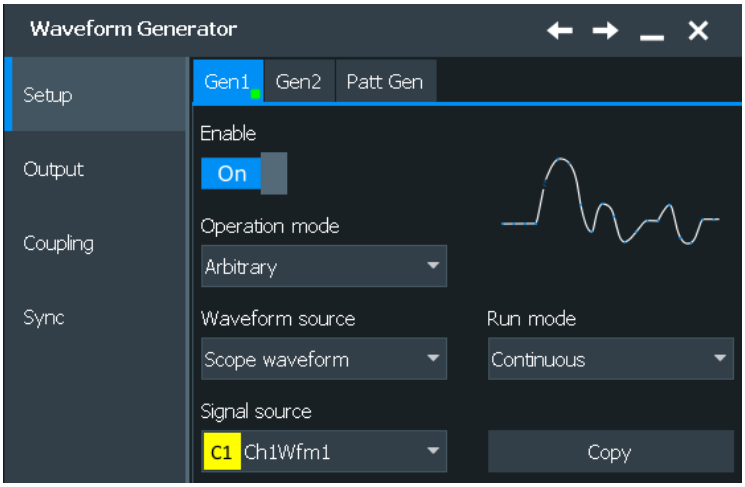
Select waveform
When a multichannel file is loaded into the arbitrary waveform generator, a dialog appears to select which waveform from the file is loaded.

Select the waveform and press "Load" to load it into the arbitrary waveform generator.



Remote command:
`WGENerator<m>:ARBGen:MULTichannel:NAME` on page 2453
`WGENerator<m>:ARBGen:MULTichannel:IMPort` on page 2453
`WGENerator<m>:ARBGen:MULTichannel:OPEN` on page 2453

15.1.5.3 Scope waveform



For a scope waveform, the following settings are available:

Signal source
Selects the oscilloscope source, from which the arbitrary signal is loaded.

Remote command:
`WGENerator<m>:ARBGen:SELect` on page 2455

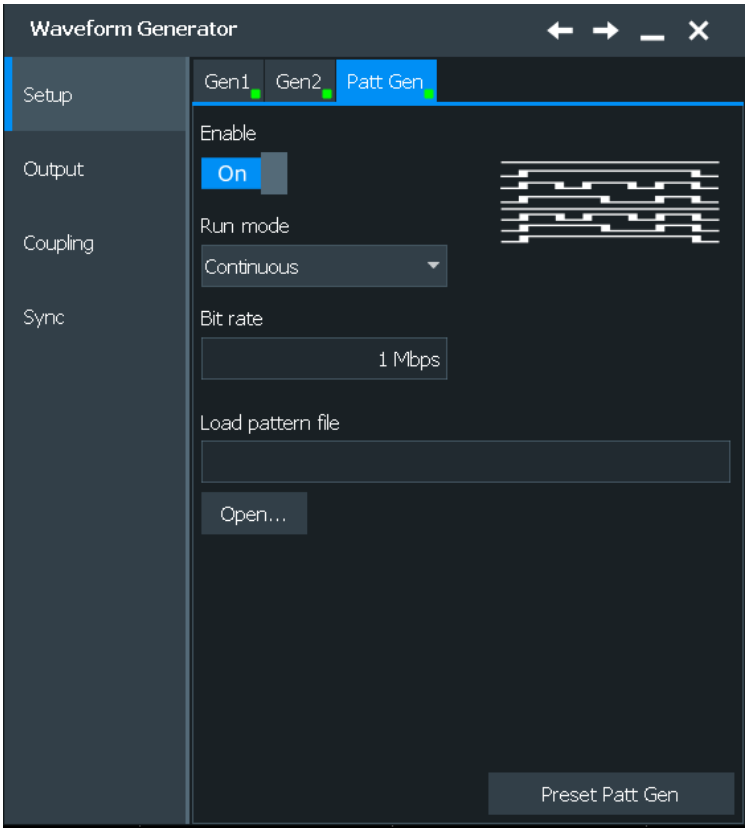
Load
Loads the waveform from the selected "Signal source".

Remote command:
`WGENerator<m>:ARBGen:COPY` on page 2453

15.2 Setup of the pattern generator

Access: [Gen] > "Setup" >"Patt Gen" tab

The pattern generator outputs parallel patterns.



NOTICE

Using pattern generator accessories

The pattern generator with connected Patt Gen Cable and Patt Gen Board (1329.7054.02) is considered as a test probe, EN 61326-2-1, clause 5.2.4.101, note 1. Therefore normal operation may display increased emissions above the limits as specified in EN 55011 and/or reduced interference resistance as required in EN 61326-1, table 1, basic requirements.

If the cable and the board are connected, other surrounding electronic devices may be disturbed. Furthermore, signals at the analog generator outputs Gen1 and Gen2 may be distorted by surrounding devices.

The settings of the pattern generator are not affected by an instrument preset. Press "Preset Patt Gen" to preset the settings of the pattern generator.

15.2.1 Content and format of the pattern generator files

The pattern generator supports `.bin` or `.csv` file formats.

Content and format of the R&S*.csv files

The file's header have the following structure:

- Bit rate [double]: the number of transmitted bits per second. The value is reflected in the user interface.
- High level [double]: the value is reflected in the user interface.
- Mask [string of 8 characters made up of "X"/"1"/"0"] : defines how the output of the pattern generator looks like. The following values are defined:
  - *1'* means that the pin output is always at high level
  - *0'* means that the pin output is always at low level (close to 0V)
  - *X* means that the pin output varies according to the given pattern
- Data sample format [HEX, BIN, OCT, DEC]: indicates how the samples are going to be interpreted. Each sample is represented as 8bit value (corresponding to the 8bit pattern generator) considering the selected format.

Example: Sample format HEX

```
Format= HEX        // Defines the format of the pattern values [HEX, DEC, BIN, OCT]
0F -> Data Sample 1
21 -> Data Sample 2
........
```

The samples are mapped on the 8 pins of the pattern generator as follows:

```
=> Pattern Samples are:
D7   D6   D5   D4   D3   D2   D1   D0
0    0    0    0    1    1    1    1
0    0    1    0    0    0    0    1
```

Example: .csv pattern generator file

```
R&S Pattern Generator File
Rate= 1000000       // Bit Rate [double]
HLevel= 1.5        // High Voltage Level [double]
Mask= X111000X     // Masks the Pins to be used in the Pattern Generator
                   // [0 => always LOW, 1 => always HIGH, X/x => used in the Pattern]
Format= DEC        // Defines the format of the pattern values [HEX, DEC, BIN, OCT]0
1
2
3
....
200
```

Content and format of the R&S pattern generator *.bin files

For the content of the fields, refer to "Content and format of the R&S*.csv files" on page 1033.

The file stream should contain the following information in the given order:

- Bit rate [double]
- High level [double]
- Mask [string of 8 characters made up of "X"/"1"/"0"]
- Number of samples [UINT32]
- Data samples [UINT8] * number of samples

15.2.2 Pattern generator settings

Enable
Enables the waveform generator/ pattern generator and outputs the signal to the connectors.

Remote command:
`WGENerator<m>[:ENABle]` on page 2443
`PGENerator:ENABle` on page 2456

Run mode
Selects the duration for which the signal of the generator will be output after the trigger event. You can choose between a "Continuous" and "Single period" duration.

Remote command:
`PGENerator:RUNMode` on page 2457

Bit rate
Sets the number of transmitted bits per second for the pattern generator.

Remote command:
`PGENerator:BITRate` on page 2456

Run single
Generates a single period of the pattern waveform if "Run mode" is set to "Single period".

Remote command:
`PGENerator:RUNSingle` on page 2457

Load pattern file
Opens a dialog for selecting an existing pattern file. It is possible to load `.bin` or `.csv` files, see Chapter 15.2.1, "Content and format of the pattern generator files", on page 1033.

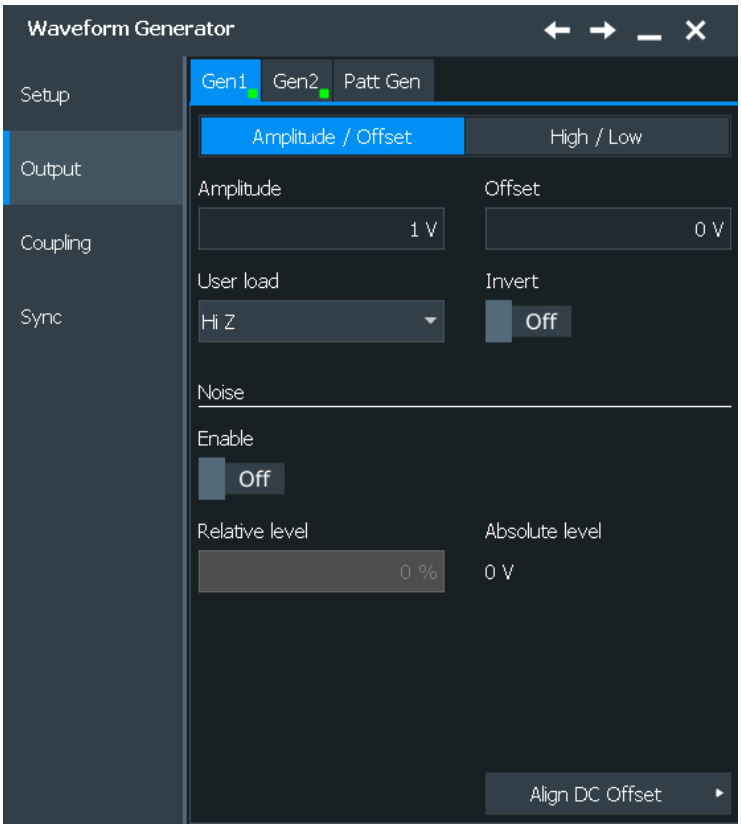
Remote command:
`PGENerator:FILE:OPEN` on page 2456
`PGENerator:FILE[:NAME]` on page 2456

Preset Gen1/Gen2/Patt Gen
Sets the parameters of the generator to their default values. The settings of the generators are not affected by an instrument preset.They are also not stored in the user-defined preset.

Remote command:
`WGENerator<m>:PRESet` on page 2443
`PGENerator:PRESet` on page 2457

15.3 Output

Access: [Gen] > "Setup" tab



In this tab you can define the generator output and add noise to the waveform.

15.3.1 Output

Amplitude/Offset - High/Low
Selects how the output is defined: with a defined "Amplitude"/ "Offset" or with "High"/"Low" values.

Amplitude
Sets the amplitude, peak to peak voltage, of the output waveform. It is defined as the voltage difference between the maximum ("High") and the minimum ("Low") voltage levels.

The "Amplitude" value is set for the currently selected "User Load". If the "User Load" is changed, the value of "Amplitude" is adapted to this new setting.

Remote command:
`WGENerator<m>:VOLTage[:VPP]` on page 2458

Offset
Sets a voltage offset.

Remote command:
`WGENerator<m>:VOLTage:OFFSet` on page 2459

High
Sets the high signal level of the output waveform.

Remote command:
`WGENerator<m>:VOLTage:HIGH` on page 2458

Low
Sets the low signal level of the output waveform.

Remote command:
`WGENerator<m>:VOLTage:LOW` on page 2459

Invert
Inverts the waveform at the offset level.

Remote command:
`WGENerator<m>:VOLTage:INVersion` on page 2459

User Load
Select the user load, the load of the DUT at its connection. You can select either a "50Ω" or a "HiZ" (high input impedance) load.

Remote command:
`WGENerator<m>:OUTPut[:LOAD]` on page 2457

DC Level
Sets the voltage DC level for the generated DC signal, for "Operation mode" >"Function Gen." and "Function type"> "DC".

Remote command:
`WGENerator<m>:VOLTage:DCLevel` on page 2458

15.3.2 Noise

Enable
Enables the adding of noise to the waveform.

Remote command:
`WGENerator<m>:MODulation:NOISe` on page 2460

Relative level
Sets the level of the noise in percentage of the set "Amplitude" output of the signal.

Remote command:
`WGENerator<m>:MODulation:NLPCent` on page 2460

Absolute level
Displays the level of the noise in volts.

Remote command:
`WGENerator<m>:MODulation:NLABsolute?` on page 2460

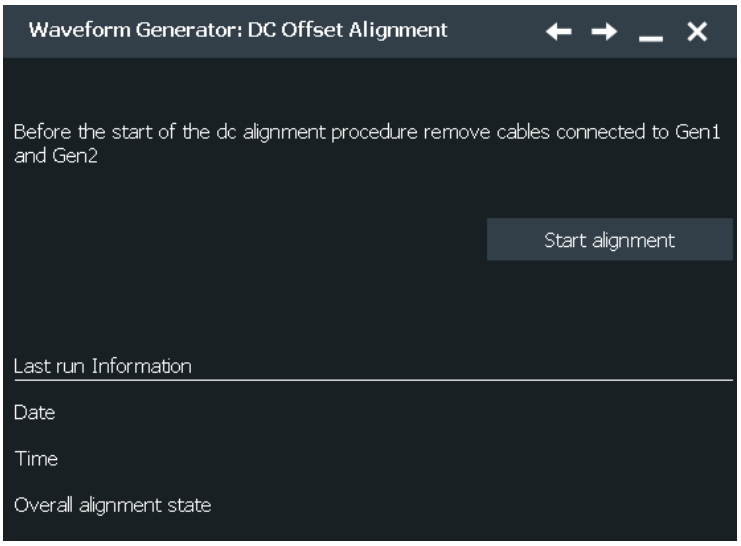
Absolute level
For "Function type">"DC" only.

Sets the level for the DC signal.

Remote command:
`WGENerator<m>:MODulation:NDCLevel` on page 2459

15.3.3 DC offset alignment

Access: [Gen] > "Output" > "Align DC Offset"



Start Alignement
Starts the alignment of the DC offset.

Remote command:
`GENerator:ALIGnment:DC[:STARt]` on page 2461

Date
Displays the date of the last performed DC offset alignment.

Remote command:
`GENerator:ALIGnment:DC:RESult:DATE?` on page 2461

Time
Displays the time of the last performed DC offset alignment.

Remote command:
`GENerator:ALIGnment:DC:RESult:TIME?` on page 2461

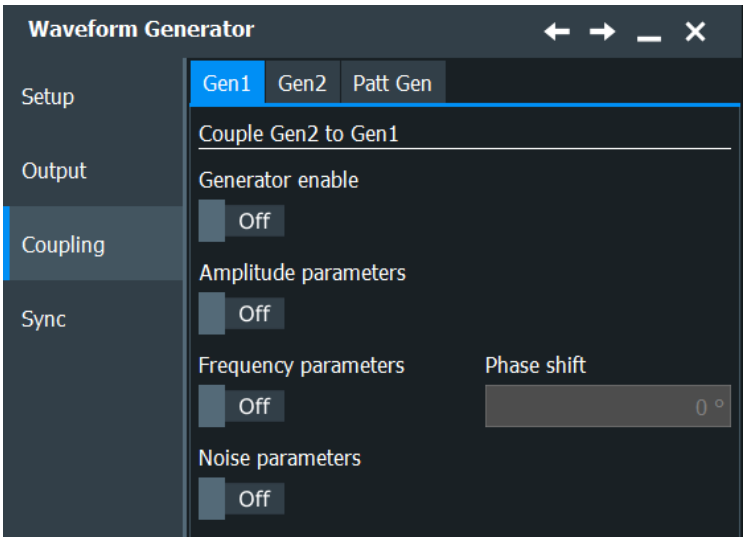
Overall alignment state
Displays the result of the DC offset alignment.

Remote command:
`GENerator:ALIGnment:DC:RESult[:STATe]?` on page 2461

15.4 Coupling settings

Access: [Gen] > "Coupling"

In the R&S RTO6, you can couple certain settings of the available waveform generators. If one of the available coupling options is enabled for one of the generators, then you cannot change the coupled parameters at the "Setup" tab of the other generator. The values for all coupled parameters are automatically taken from the currently selected generator.



Generator enable
Enables the coupling of the generators, with the selected set of parameters: amplitude, frequency and noise.

Remote command:
`WGENerator<m>:COUPling:STATe` on page 2462

Amplitude parameters
Couples all amplitude parameters of "Gen1" to "Gen2" or "Gen2" to "Gen1" including the load. The values are taken from the currently selected generator.

- For all "Operation Mode": the output settings except of "Load" and "Inversion", "Amplitude", "High", "Offset" and "Low".
- For "Operation Mode" > "Function Gen.": "DC Level"
- For "Operation Mode" > "Modulation": "Modulation depth"

Remote command:
`WGENerator<m>:COUPling[:FREQuency]` on page 2463

Frequency parameters
Couples all frequency parameters of "Gen1" to "Gen2" or "Gen2" to "Gen1". The values are taken from the currently selected generator.

- For "Operation Mode" > "Function Gen.": "Frequency" and "Period"
- For "Operation Mode" > "Modulation": "Carrier frequency", "Carrier period", "Modulation freq", "Freq deviation", "Frequency 1", "Frequency 2" and "Hop rate"
- For "Operation Mode" > "Sweep": "Start frequency", "Stop frequency" and "Sweep time".
- For "Operation Mode" > "Arbitrary": "Arb wfm source" and "Signal source".

You can still change the other settings of the generators independently.

Remote command:
`WGENerator<m>:COUPling:AMPLitude` on page 2462

Phase shift
Sets the phase shift between the waveform of "Gen1" and "Gen2" when the frequency parameters of the two waveforms are coupled.

Remote command:
`WGENerator<m>:COUPling:PHASeshift` on page 2462

Noise parameters
Couples all noise parameters of "Gen1" to "Gen2" or "Gen2" to "Gen1". The values are taken from the currently selected generator.

Remote command:
`WGENerator<m>:COUPling:NOISe` on page 2463

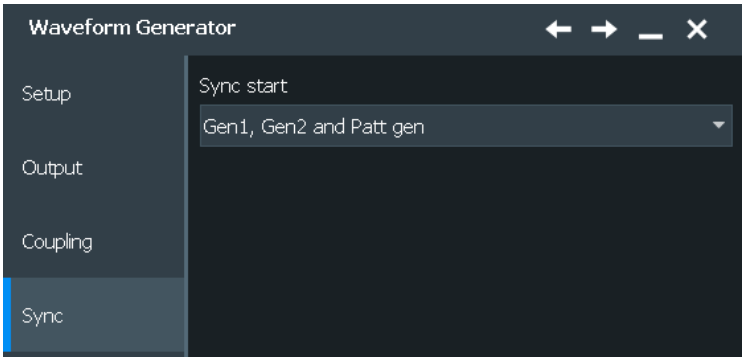
Arbitrary waveform
Available for General settings = "Arbitrary".

Couples the arbitrary waveform "Gen1" to "Gen2" or "Gen2" to "Gen1". The values are taken from the currently selected generator.

Remote command:
`WGENerator<m>:COUPling:NOISe` on page 2463

15.5 Sync settings

Access: [Gen] > "Sync"



Sync start
Selects, which signals generated from the waveform generator are synchronized.

Selecting one of the sync options indicates that the first samples of those signals are generated at the same time, irrespective of if the generators are on or off. Selecting one of the coupling options automatically syncs the signals generated by the two waveform generators.

Remote command:
`GENerator:SYNC[:COMBination]` on page 2463

15.6 Configuring the waveform generator

This chapter explains step-by-step how to configure the waveform generator.

- Configuring a function waveform....1040
- Configuring a modulation waveform....1041
- Configuring a sweep waveform....1042
- Configuring an arbitrary waveform....1042
- Configuring a pattern generator waveform....1043

15.6.1 Configuring a function waveform

1. Press the [Gen] key on the front panel.
2. Select the "Setup" tab.
3. Under "Operation mode", enable the "Function Gen." button.
4. Select the "Function type" that you want to generate, e.g. "Sine".
5. Depending on the selected "Function type", configure the settings of the waveform like "Frequency" and "Period".
6. If necessary, change the "Output" settings or add "Noise" to the waveform.

7. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

15.6.2 Configuring a modulation waveform

Generating an AM modulated waveform

1. Press the [Gen] key on the front panel.
2. Select the "Setup" tab.
3. Under "Operation mode", enable the "Modulation" button.
4. Tap "Modulation type" and select "AM".
5. Set the "Carrier frequency" and the "Carrier period".
6. Tap "Modulation signal" and select the required waveform.
7. Depending on the selected "Modulation signal", configure the settings of the waveform like "Modulation freq" and "Modulation depth".
8. If necessary, change the "Output" settings or add "Noise" to the waveform.
9. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

Generating an PWM modulated waveform

1. Press the [Gen] key on the front panel.
2. Select the "Setup" tab.
3. Under "Operation mode", enable the "Modulation" button.
4. Tap "Modulation type" and select "PWM".
5. Set the "Carrier frequency" and the "Carrier period".
6. Tap "Modulation signal" and select the required waveform.
7. Depending on the selected "Modulation signal", configure the settings of the waveform like "Modulation freq" and "Modulation depth".
8. If necessary, change the "Output" settings or add "Noise" to the waveform.
9. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

Generating an FM modulated waveform

1. Press the [Gen] key on the front panel.
2. Select the "Setup" tab.
3. Under "Operation mode", enable the "Modulation" button.

4. Tap "Modulation type" and select "FM".
5. Set the "Carrier frequency" and the "Carrier period".
6. Tap "Modulation signal" and select the required waveform.
7. Depending on the selected "Modulation signal", configure the settings of the waveform like "Modulation freq" and "Freq deviation".
8. If necessary, change the "Output" settings or add "Noise" to the waveform.
9. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

Generating an FSK modulated waveform

1. Press the [Gen] key on the front panel.
2. Select the "Setup" tab.
3. Under "Operation mode", enable the "Modulation" button.
4. Tap "Modulation type" and select "FSK".
5. Set the "Frequency 1", "Frequency 2" and the "Hop rate".
6. If necessary, change the "Output" settings or add "Noise" to the waveform.
7. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

15.6.3 Configuring a sweep waveform

1. Press the [Gen] key on the front panel.
2. Select the "Setup" tab.
3. Under "Operation mode", enable the "Sweep" button.
4. Set the "Start frequency", the "Stop frequency" and the "Sweep time".
5. If necessary, change the "Output" settings or add "Noise" to the waveform.
6. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

15.6.4 Configuring an arbitrary waveform

Generating an arbitrary waveform from a saved file

1. Press the [Gen] key on the front panel.
2. Select the "Setup" tab.

3. Under "Operation mode", enable the "Arbitrary" button.
4. Set the "Arb wfm source" to "Saved waveform".
5. Select the "Run mode".
6. Press "Open" and set the path to your saved arbitrary waveform.
7. Set the "Sample rate".
8. If necessary, change the "Output" settings or add "Noise" to the waveform.
9. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

Generating an arbitrary waveform from the scope waveform

1. Press the [Gen] key on the front panel.
2. Select the "Setup" tab.
3. Under "Operation mode", enable the "Arbitrary" button.
4. Set the "Arb wfm source" to "Scope waveform".
5. Select the "Run mode".
6. Press the "Signal source" button and select the channel source for the waveform.
7. Set the "Sample rate".
8. If necessary, change the "Output" settings or add "Noise" to the waveform.
9. Press the "Enable" button, to output the waveform at the output connector of the waveform generator.

15.6.5 Configuring a pattern generator waveform

1. Press the [Gen] key on the front panel.
2. Select the "Setup" tab.
3. Select the "Patt Gen" tab.
4. Select the "Run mode".
5. Press "Open" and set the path to your saved pattern file.
6. Set the "Bit rate" and "High level".
7. Press "Open" and set the path to your saved arbitrary waveform.
8. Press the "Enable" button, to output the pattern at the output connector of the pattern generator.

16 Pulse source and TDR/TDT analysis (options R&S RTO6-B7/K130)

16.1 Pulse source (option R&S RTO6-B7)

The pulse source option R&S RTO6-B7 provides a symmetrical differential pulse signal with steep rise time and configurable parameters. The hardware module of the option features four 2.92 mm connectors (K type), two for signal output (Out, $\overline{\text{Out}}$), and two for reference signal output (Ref, $\overline{\text{Ref}}$).

Typical applications of the pulse source are:

- Stimulus for devices under test, for example, as a precise clock
- Accurate source for deskewing cables and probes for differential measurements
- Analysis of time-transmission characteristics of a differential signal path
- TDR/TDT analysis with option R&S RTO6-K130

The module is installed at the rear panel of the instrument. To ensure correct installation and calibration, send the instrument to a Rohde & Schwarz service center.

Preventing electrostatic discharge (ESD)

Electrostatic discharge is most likely to occur when you connect or disconnect a DUT.

► **NOTICE!** Electrostatic discharge can damage the electronic components of the product and the device under test (DUT).

Ground yourself to prevent electrostatic discharge damage:

a) Use a wrist strap and cord to connect yourself to ground.
b) Use a conductive floor mat and heel strap combination.

Usage of cables

Beside the pulse source module, the option includes several accessories: SMA cables, 50 Ω terminations, adapters, and more. The accessory case provides also space for optional accessories. All included accessories are listed in the data sheet.

NOTICE

When fastening the cable, observe the maximum torque of 0.9 Nm. Therefore, using a 0.9 Nm torque wrench is recommended (optional accessory 1328.8534.35).

To ensure accurate and reliable measurements, connect only cables with a maximum length of 3 m.

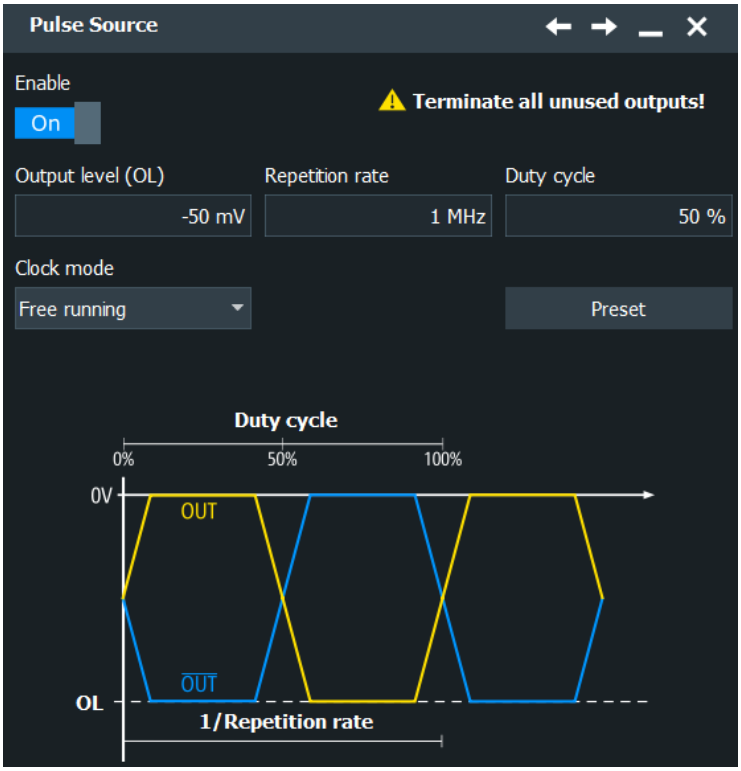
Always use double-shielded cables to prevent electromagentic interference (EMI) from impairing measurement results.

Always use cables to connect the DUT or test fixture to the pulse source outputs. Never connect the DUT or test fixture directly to the outputs.

16.1.1 Settings for the pulse source

Access: [App Cockpit] > "Generators" > "PSC"

Terminate all unused outputs of the pulse source with 50 Ω terminating resistors.



Enable
Switches all four pulse source outputs on and off.

Remote command:
`PSRC[:STAT]` on page 2464

Output level
Sets the low level of the output pulse. The high level is 0 V and cannot be changed.

Remote command:
`PSRC:OUTPutlow` on page 2464

Repetition rate
Sets the pulse frequency, the repetition rate of the pulse. Available values depend on the selected clock mode.

Remote command:
`PSRC:REPRate` on page 2465

Duty cycle
Sets the duty cycle of the pulse in %, which is the ratio of the positive pulse width to the period of the signal.

For repetition rates > 5 MHz, the value is fixed at 50 %.

Remote command:
`PSRC:DUTYcycle` on page 2465

Clock mode
Sets the dependency of the pulse clock on the instrument's reference clock. The appropriate clock mode depends on the measurement application.

"Free running" The clock of the pulse source is independent. Deviations of the system do not affect the pulse clock, and deterministic conditions are avoided.
If the waveform aquisition is set to "Average", the "free running" clock mode can improve the measurement result. Internal spurious signals of the instrument are reduced by averaging, because they are not coherent to the pulse source signal.
See also: "Arithmetic" on page 130.

"Locked" The pulse source is locked to the reference clock of the instrument.

Remote command:
`PSRC:CLOCkmode` on page 2465

Preset
Resets the pulse source to the default state. Pulse source settings are not are not affected by an instrument preset. They are also not stored in the user-defined preset.

Remote command:
`PSRC:RST` on page 2465

16.2 TDR/TDT analysis (option R&S RTO6-K130)

The option R&S RTO6-K130 provides analysis of time domain reflectometry (TDR) and time domain transmission (TDT) to verify high-speed digital designs. Using the results of the TDR/TDT calibration, the instrument removes the effects of the cables (deembedding), and measures the DUT.

Typical use cases are:

- Locate faults in a circuit
- Measure impedance
- Measure step response of a DUT

16.2.1 About TDR/TDT measurements

Using TDR/TDT measurements, you can analyze the reflections and transmission characteristics of a DUT. With TDR, it is possible to determine the type, magnitude and location of a fault. TDT delivers information on stability, losses and bandwidth of the DUT.

The signal source is the pulse source, and the oscilloscope measures the reflected and transmitted signals.

Requirements

TDR/TDT analysis requires the pulse source option R&S RTO6-B7.

Cabling requirements:

- Use only high-quality double-shielded RF cables. Two suitable cables are delivered with the pulse source option R&S RTO6-B7.
- Use low loss cables with frequency range that fits the oscilloscope bandwidth.
- Consider the recommended cable length: 0.5 m < length < 3 m.

Calibration standards

During calibration, you measure the connecting cables of the measurement setup against standards. Calibration standards assumed to be ideal.

- Short with RF quality: a standard short is delivered with the pulse source option R&S RTO6-B7. You can use other shorts with electrical length 0 ns … 2 ns. Default value is the R&S RTO6-B7 short.
- Open (DC)
- Match (DC): a standard 50 Ω match is delivered with the pulse source option R&S RTO6-B7.
  You can use another match with 50 Ω±10%.
- Through (RF): the length is assumed to be zero.

Calibration results

After successful calibration, the frequency response of the short and through measurements are displayed. Steep dips in the graph indicate high attenuation within your desired bandwidth, which impairs the performance. To improve the measurement, reduce the bandwidth of the system. To check the impact of the frequency response calibration on the time domain waveform, you can disable the usage ("Use frequency response calibration"). Without frequency response correction, only the rescaling to the TDR/TDT measurements is performed.

The calibration results are used for correction of subsequent measurements and can be saved in a file for later usage.

Correction

Calibration results are used for correction of the signal in two ways:

- DC scaling using results of open, short and match calibration.
- RF correction and deembedding of transmission loss based on short calibration.

The R&S RTO6 corrects for the transmission loss of the system, but not for multiple reflections between DUT and oscilloscope.

16.2.2 Using TDR/TDT

16.2.2.1 Calibrating the setup

Before you can start TDR/TDT measurements, calibrate the used cables.

You can:

- Calibrate the setup using the calibration wizard of the R&S RTO6.
- Load a file with results of a previous calibration of the same setup.
- Use default calibration and correction values.

To calibrate the cables

The calibration procedure is described for single-ended TDR and TDT analysis. If you need only TDR or only TDT, the procedure is the same but with fewer calibration steps and fewer measurements.

1. Open [App Cockpit]> "Analysis" > "TDR / TDT"
2. To start a new calibration, select "Start wizard".

   If a calibration was loaded or done before, the calibration steps are skipped, and the measurement dialog box is shown.
3. Select the analysis type: "TDR", "TDT", or "TDR / TDT" (both TDR and TDT measurements).
4. Select "Calibrate setup".
5. Connect the cables, and select the input channels:
   a) Connect the first cable to the "Ref" connector of the pulse source and to a channel input. This cable is for TDR measurements.
   b) Select the connected channel in the dialog box.
   c) Connect the second cable to the "Out" connector. Do not connect the DUT.
   d) Connect a third cable to a channel input. This cable is for TDT measurements.
   e) Select the connected channel in the dialog box.
   f) Terminate all unused connectors of the pulse source.

g) Set up the input channels, in particular, the trigger level and vertical settings. These settings must remain unchanged during and after calibration.
h) Select "Next".

6. Calibrate with a short:
   a) Connect a short to the cable that is connected to the "Out" connector.
   b) To change the short, select the edit icon.
   c) If you use another short than the delivered standard short, select "User.defined", and enter the electrical length of the used short.
   d) Select "Calibrate".
7. Calibrate with open cable:
   a) Remove the short.
   b) Select "Calibrate".
8. Calibrate with a match:
   a) Connect a match to the cable that is connected to the "Out" connector.
   b) Select "Calibrate".
9. Calibrate with a through:
   a) Connect a through to the cable that is connected to the "Out" connector.
   b) Connect the through to the third cable.
   c) Select "Calibrate".
10. Select "Next".
11. Check the graphical calibration result. To see the details, you can display the full-size graphic, zoom in and out, and use the cursor. To see the original plot again, select "Reset".

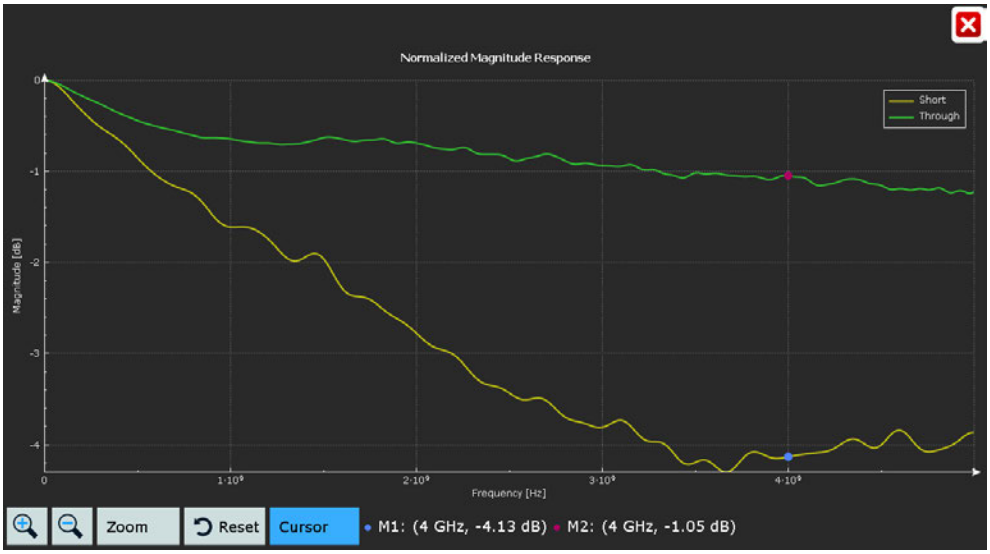


Figure 16-1: Full-size display of calibration results, with cursor

12. Save the calibration results for later measurements.

13. Select "Next".

    Now you can analyze the DUT.

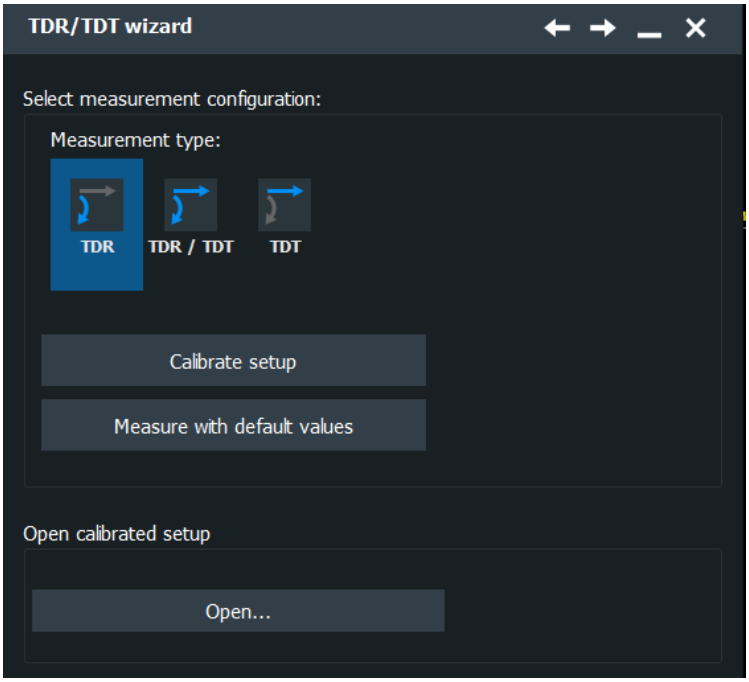
16.2.2.2 Measuring the DUT

1. Open [App Cockpit]> "Analysis" > "TDR / TDT"
2. Load an existing calibration file, or calibrate the setup as described in Chapter 16.2.2.1, "Calibrating the setup", on page 1048.
3. Discharge the DUT before connecting.
4. Connect the DUT instead of calibration standards as shown in the dialog.
5. Under "Signals", select the characteristics that you want to analyze.
6. Enter the "Bandwidth" of the DUT.
7. Select whether you want to see the time or the distance on the x-axis: "Domain".
8. If the distance is selected, enter the permittivity of the DUT: "ε_r".
9. To check the impact of the frequency response calibration on the time domain waveform, disable "Use frequency response calibration".
10. Close the dialog box.
11. Set the horizontal time scale so that the complete DUT is captured and visible on the screen.
12. Analyze the selected signals. See also: Chapter 16.2.6, "Measurement results", on page 1057

16.2.3 Calibration settings

16.2.3.1 Setup: TDR/TDT configuration

Access: [App Cockpit]> "Analysis" > "TDR / TDT"



Measurement type
Selects the analysis type: "TDR" (time domain reflectometry), "TDT" (time domain transmission), or "TDR / TDT" (both TDR and TDT).

Remote command:
`TDRT:TYPE` on page 2467

Calibrate setup
Starts the wizard for manual calibration. In the next steps, you connect several calibration standards and measure the correction parameters.

Measure with default values
Starts the measurement with default correction values instead of calibrating the setup. Default values are rough values without time reference.

Remote command:
`TDRT:DEFault` on page 2466

Open
Opens a saved calibration file of a previous calibration and starts the measurement.

Remote command:
`TDRT:CALibration:OPEN:NAME` on page 2466
`TDRT:CALibration:OPEN[:EXECute]` on page 2466

16.2.3.2 Setup: channels

Access: [App Cockpit]> "Analysis" > "TDR / TDT" > "Calibrate setup"

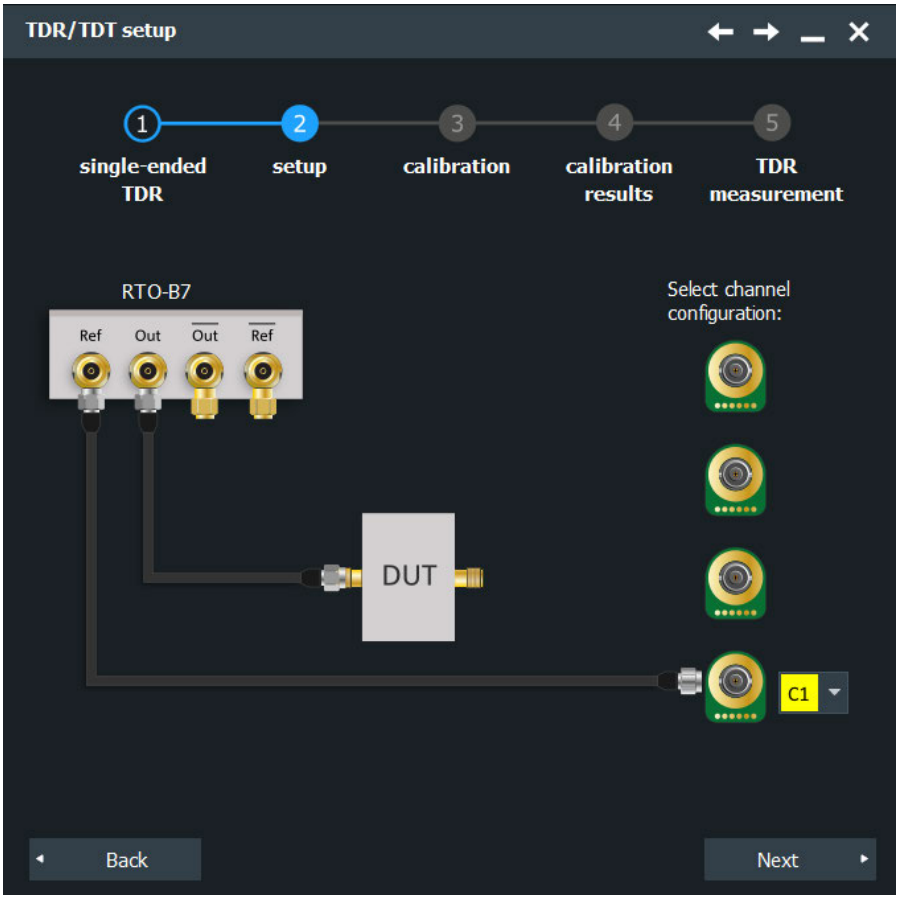


Figure 16-2: Channel setup for single-ended TDR/TDT

Channel
Select the input channels that are connected to the "Ref" connector and the "Out" connector of the pulse source.

Note: The position of the DUT is shown for information only. Do not connect the DUT.

Remote command:
`TDRT:PULSe:OUT` on page 2467
`TDRT:PULSe:REFerence` on page 2467

16.2.3.3 Calibration

Access: [App Cockpit]> "Analysis" > "TDR / TDT" > "Calibrate setup" > "Next"

Figure 16-3: Calibration of single-ended TDR/TDT

The used short type is shown in the wizard. To change the short, tap the "Edit" icon.

Short type
Selects whether the short is the Rohde & Schwarz short that is delivered with the pulse source R&S RTO6-B7, or another one.

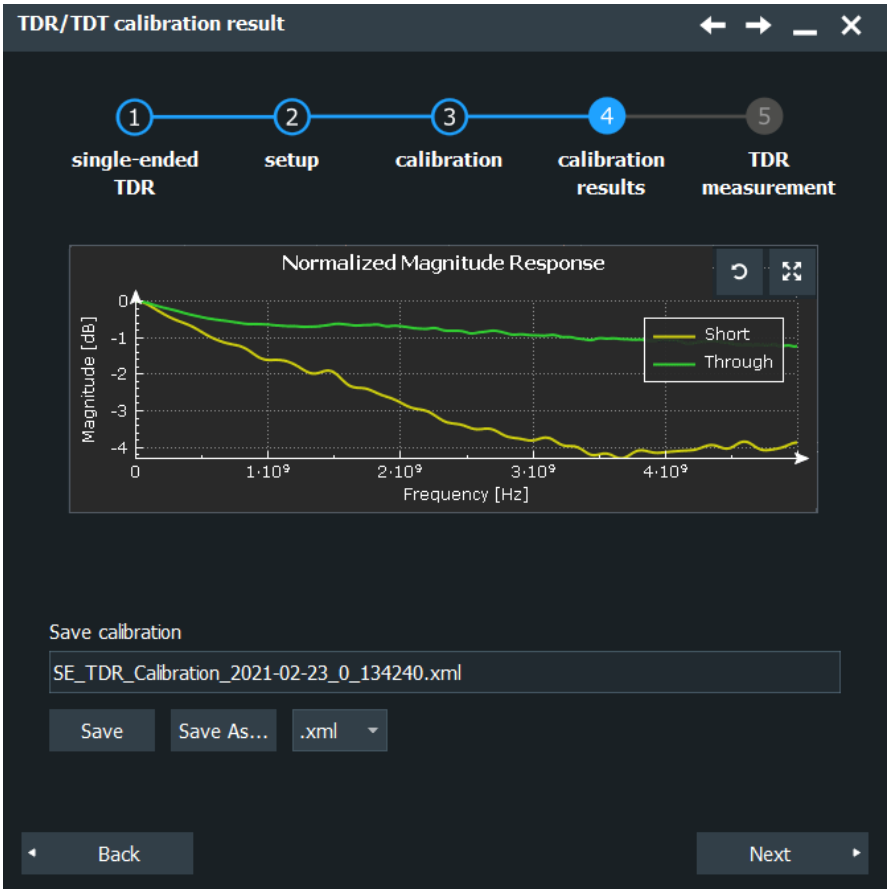
If you do not use the Rohde & Schwarz short, enter the "Electrical short length" of the used short.

Remote command:
`TDRT:SLENgth` on page 2467

Calibrate
Starts the calibration of the current calibration step (short, open, match through).

16.2.4 Calibration results

Access: [App Cockpit]> "Analysis" > "TDR / TDT" > "Calibrate setup" > "Next"



Magnitude response plot

Shows the graph of magnitude over frequency for the short calibration (TDR) and through calibration (TDT).

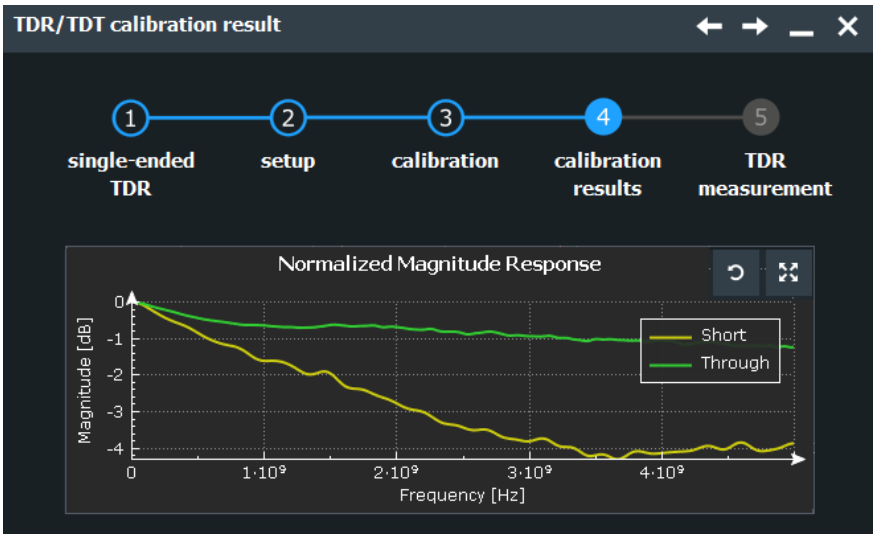


Figure 16-4: Result of TDR and TDT calibration: smooth graph with moderate attenuation

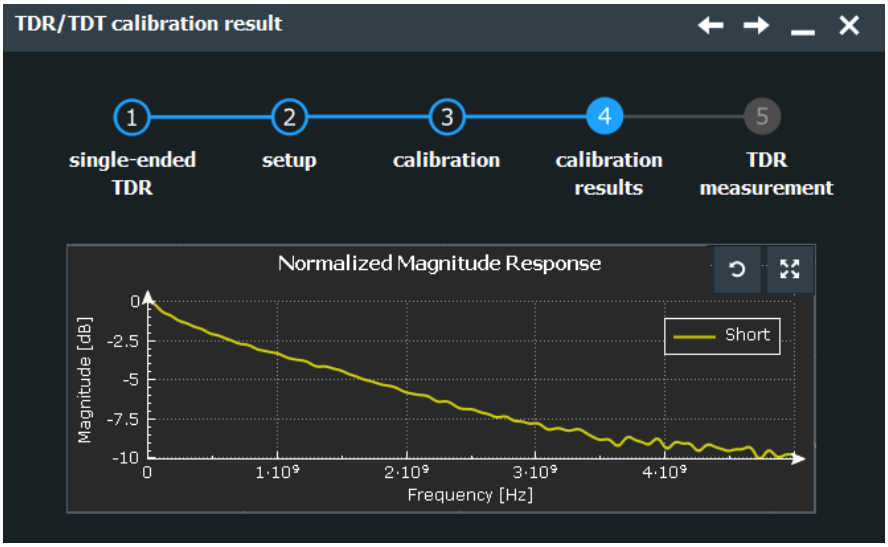


Figure 16-5: Result of TDR calibration: high attenuation and noisy graph at higher frequencies

Save calibration

Saves the current correction values to a file. Using the saved file, you can repeat the measurements at any time.

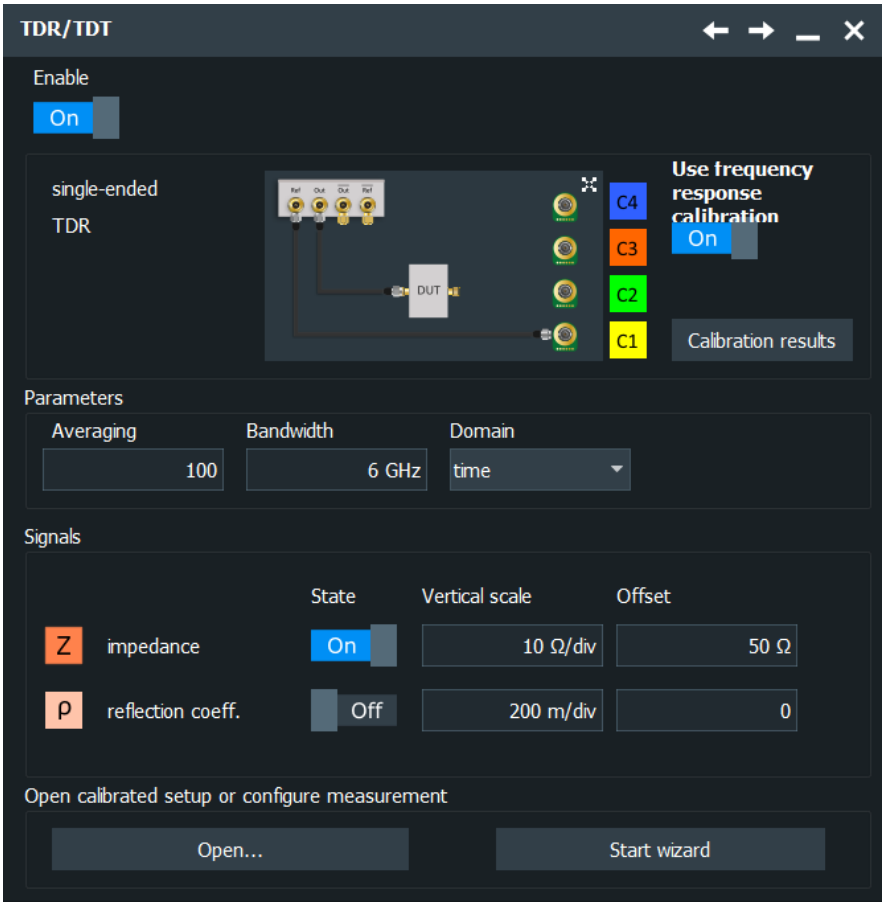
Remote command:

`TDRT:CALibration:SAVE:NAME` on page 2468

`TDRT:CALibration:SAVE[:EXECute]` on page 2468

16.2.5 Measurement settings

Access: [App Cockpit]> "Analysis" > "TDR / TDT" > "Calibrate setup" > "Next"> "Next" > "Next"



Enable
Activates the TDR/TDT measurements.

Remote command:
`TDRT:STATe` on page 2468

Use frequency response calibration
Disables the frequency response calibration. Without calibration, the measured signals are only rescaled, and you can see the impact of the frequency response calibration on the time domain waveform.

The setting is not available when measuring with default values.

Remote command:
`TDRT:CALibration:USECaldata` on page 2468

Averaging
The "Averaging" value is the "Average count (N-single count)", which you can also set in the acquisition and trigger settings. See "Average count (N-single count)" on page 127.

Increase the value to reduce noise.

Bandwidth
Sets the bandwidth of the DUT.

Remote command:
`TDRT:BW` on page 2469

Domain, ε_r
Sets the domain of the x-axis: time or distance (in m or inch). If you select distance, enter also the permittivity "ε_r" of the used DUT.

Remote command:
`TDRT:DOMain` on page 2469
`TDRT:PERMittivity` on page 2469

State
Select the TDR/TDT signals to be displayed and analyzed.

TDR delivers the impedance Z and the reflection coefficient ρ.

TDT delivers the normalized and corrected step response.

Remote command:
`TDRT:SIGNal<m>STATe` on page 2469

Vertical scale
Set the vertical scale of the TDR/TDT signals.

Remote command:
`TDRT:SIGNal<m>:SCALe` on page 2470

Offset
Set the offset of the TDR/TDT signals.

To set the position, assign position to the vertical Position knob ("File" menu > "Front-panel Setup" > "Knobs"). Turn the vertical Position knob to change the position.

Remote command:
`TDRT:SIGNal<m>:OFFSet` on page 2470
`TDRT:SIGNal<m>:POSition` on page 2470

16.2.6 Measurement results

To get exact measurement results, set the horizontal time scale so that the complete DUT is captured and visible on the screen. Thus, interference between the reflected signals from the DUT is avoided. Interference can occur because the repetition rate of the pulse source is linked to the horizontal scale to maximize measurement speed.

The results of the TDR measurements are the impedance waveform Z and the waveform of the reflection coefficient ρ.

To investigate details of the waveforms, use the zoom function. For further analysis, you can use automatic and cursor measurements, histogram, mathematics and reference waveforms.

Data export

In the same way as channel waveforms, you can export the data of Z, ρ and H waveforms to file (one waveform per file). Raw data is not available.

See also: Chapter 12.2, "Save and recall waveform data and results", on page 452.

Remote commands:

- `EXPort:WAVeform:SOURce` on page 1649
- `TDRT:SIGNal<m>:DATA:HEADer?` on page 2471
- `TDRT:SIGNal<m>:DATA:VALues?` on page 2471

16.2.7 Troubleshooting

If the calibration or measurement does not show meaningful results, the following table helps you to find the cause.

	Issue	Checks
Calibration	No waveform is visible.	Check the cabling. Check the pulse source.
Calibration	Calibration with short failed	Check the cable length: 0.5 m < length < 3 m Use cables with low loss. Use a short that has a low impedance for high frequencies.
Calibration	Calibration with open failed	Use cables with low loss.
Calibration	Calibration with match failed	Use cables with low loss. Use match with 50 Ω±10% for single-ended measurements.
Calibration	Calibration with through failed	Use cables with low loss. Check the cable length: 0.5 m < length < 3 m
Calibration	Failure after completion of all calibration steps	Use cables with low loss. Check the pulse source.
Measurement	Waveform is noisy or unstable	Evaluate the calibration results for any resonances or high attenuation in frequency response. Increase averaging. Reduce bandwidth. Disable calibration. If noise disappears, reduce attenuation or remove resonances from your setup and recalibrate.

17 I/Q software interface (Option R&S RTO6-K11)

The option R&S RTO6-K11 "I/Q software interface" acquires modulated signals and outputs I/Q data for further analysis using other software, for example, MATLAB. Rohde & Schwarz provides the R&S®VSE vector signal explorer for analysis of I/Q data on external computer.

In I/Q mode, the input signal is captured with a sample rate of 10 Gsample/s.

In addition to the I/Q data recording, the option also provides the NFC trigger type, a specific trigger for near field communication analysis.

For specifications, see the R&S RTO6 data sheet.

- I/Q mode 1059
- I/Q sampling setup 1059
- I/Q data output 1064
- NFC trigger 1066

17.1 I/Q mode

The I/Q mode is a special operating mode of R&S RTO6. In this mode, the usual analyzing tools (cursor, zoom, measurements, FFT and so on) of the oscilloscope are deactivated. The instrument displays the magnitude of the I/Q vector to get an impression of the data before it is exported.

Triggering is performed on the input data before conversion to I/Q data, using the usual R&S RTO6 trigger functionality. In case of complex input signals, the instrument triggers on one component of the complex signal (I or Q) depending on the selected trigger source.

To activate the IQ mode

1. Press the [App Cockpit] key on the front panel.
2. In the "RF" tab, select "IQ".
3. In the "IQ Sampling" dialog, tap "Enable".

   The instrument switches to I/Q mode, and the setup dialog for I/Q data recording is shown.

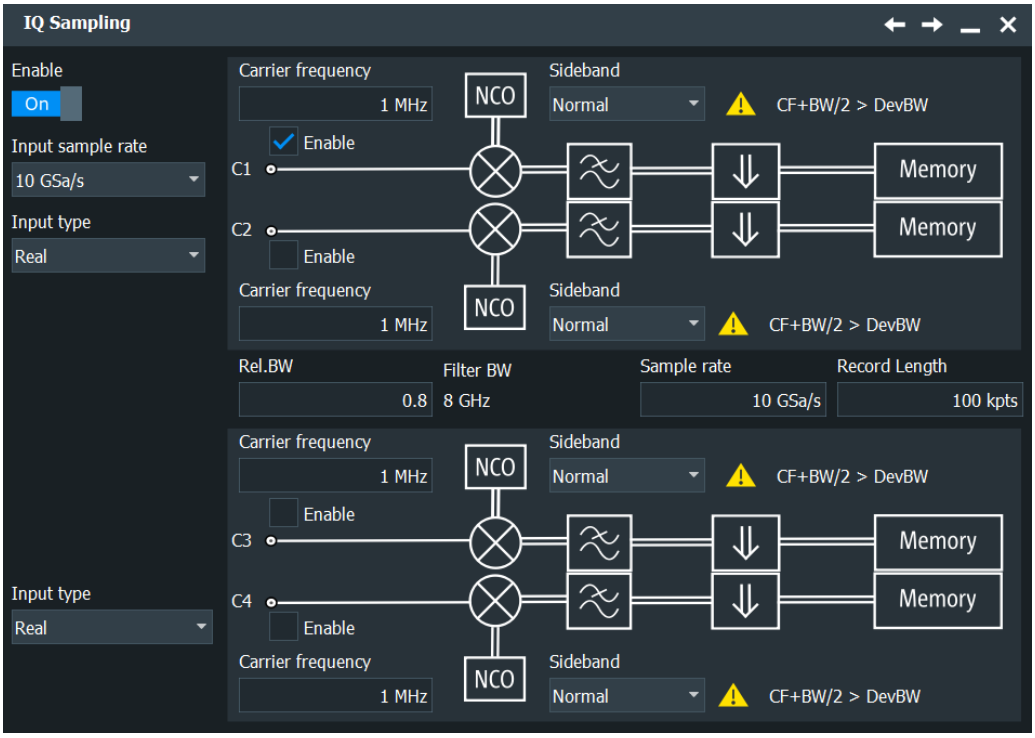
17.2 I/Q sampling setup

Access: [App Cockpit] menu > "RF" tab > "IQ" > "IQ Sampling Setup"

In the "IQ Sampling" dialog box, you configure the complete I/Q sampling for real and complex input signals.

If the values in the dialog box result in an unfavorable combination, a note is shown. The note indicates:

- "Undersampling" if the carrier frequency is too fast.
- "CF + BW/2 > DevBW", where CF is the carrier frequency, BW is the filter bandwidth, and DevBW is the maximum instrument bandwidth.
- Remote commands:
  `CHANnel<m>:IQ:WARNing[:STATe]?` on page 2476
  `CHANnel<m>:IQ:WARNing:MESSage?` on page 2476



As usual, the signal icons of the enabled channels show the current settings. In I/Q mode, the I/Q settings are shown, see Figure 17-1. The left signal icon shows a complex input signal in IF range with carrier frequency 1 MHz and normal/upper sideband. C1 is the in-phase component, C2 the quadrature component. The right signal icon shows a real input signal with carrier frequency 1 MHz. C1 has inverse sideband, C2 has normal sideband.

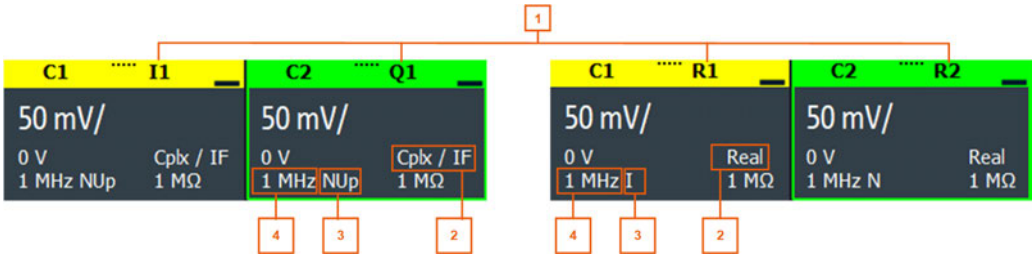


Figure 17-1: Signal icons in I/Q mode

1 = Signal component
2 = Input type/Input mode
3 = Sideband
4 = Carrier frequency

Enable
Enables the I/Q mode to record I/Q data.

The analyzing tools on the toolbar are deactivated, and the magnitude of the I/Q vector is displayed.

Remote command:
`IQ:STATe` on page 2473

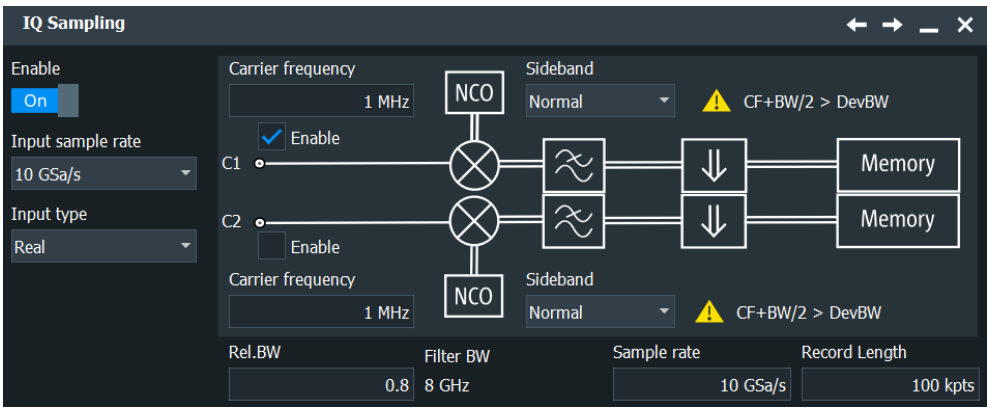
Input sample rate
Shows the sample rate of capturing the input data in I/Q mode: 10 Gsample/s.

Remote command:
`IQ:INSRate` on page 2473

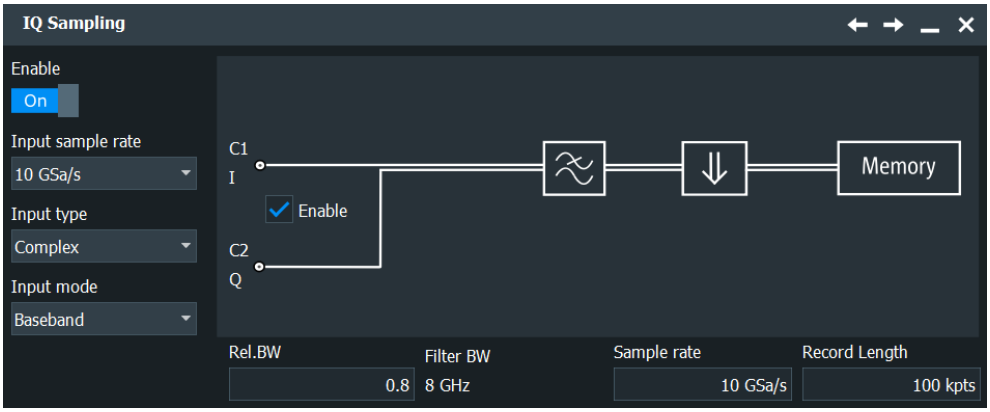
Input type, Input mode
"Input type" sets the format of the input signal. "Input mode" selects the frequency band of a complex input signal.

The R&S RTO6 I/Q software interface supports three formats of input signals.

- Real RF signals

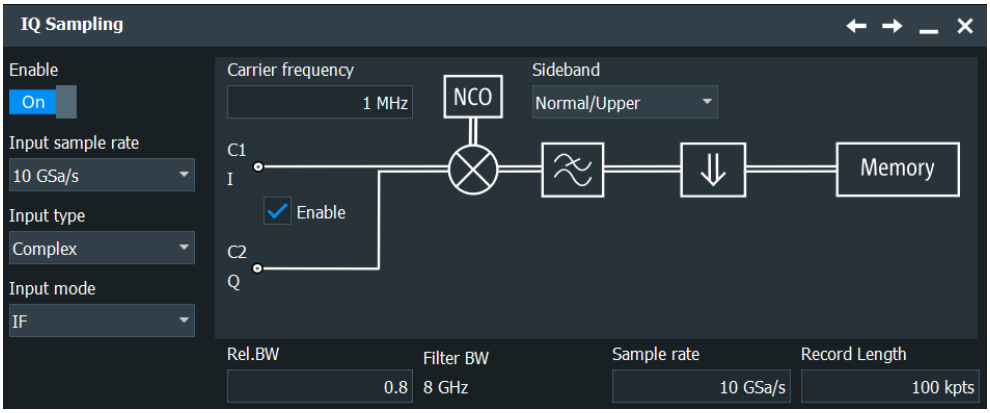


  - The input signal is down-converted, filtered and resampled to the sample rate of the output I/Q signal.
  - One channel is required for each input signal. Thus up to four real signals can be recorded in parallel.
  - Sideband settings: see "Sideband (real input)" on page 1063.
- Complex I/Q baseband signals



– The input signal is filtered and resampled to the sample rate of the output I/Q signal.
– Two input channels are required for each input signal, one for the in-phase component, and one for the quadrature component. Thus up to two complex input signals can be recorded in parallel.

- Complex I/Q signals in low IF range



– The input signal is down-converted, filtered and resampled to the sample rate of the output I/Q signal.
– Two input channels are required for each input signal, one for the in-phase component, and one for the quadrature component. Thus up to two complex input signals can be recorded in parallel.
– Sideband settings: see "Sideband (complex IF input)" on page 1064.

Remote command:
`CHANnel<m>:IQ:INPType` on page 2473
`CHANnel<m>:IQ:INPMode` on page 2473

Rel. BW, Filter BW

"Rel. BW" sets the bandwidth factor to define the filter bandwidth:

*Filter BW = Relative BW * Sample rate*

Remote command:
`IQ:RBWidth` on page 2474
`IQ:BWIDth?` on page 2474

Sample rate
Sets the required sample rate of the output I/Q data.

Remote command:
`IQ:SRATe` on page 2474

Record length
Sets the required record length of the output I/Q data. The resulting acquisition time of the I/Q data is:

Acquisition time = Record length / Sample rate

Remote command:
`IQ:RLENgth` on page 2474

Channel On/Off
Switches the channel signal on or off. The signal icon appears on the signal bar. The waveform of the last acquisition is displayed in the diagram.

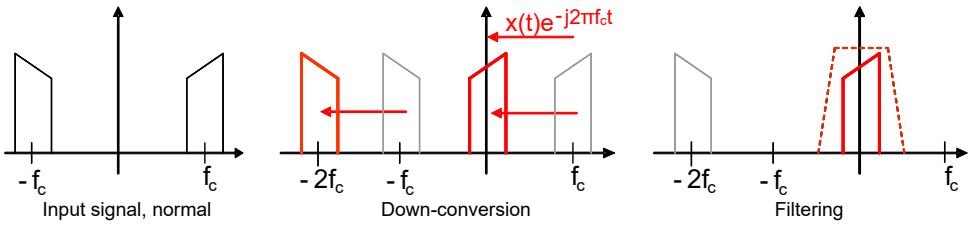
Remote command:
`CHANnel<m>:STATe` on page 1337

Carrier frequency
Sets the carrier frequency of the modulated RF signal or of the complex signal in IF range. The minimum carrier frequency is 10 mHz, the maximum frequency corresponds to the input sample rate.

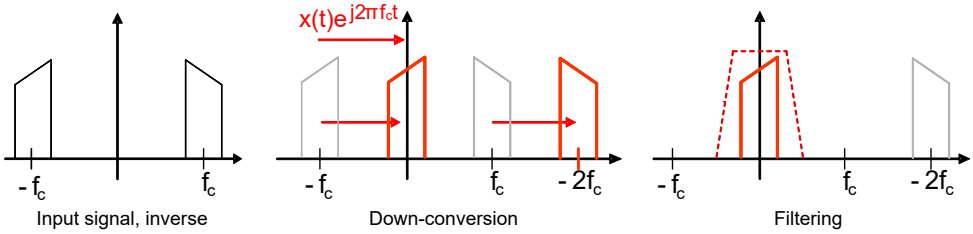
Remote command:
`CHANnel<m>:IQ:CFRequency` on page 2475

Sideband (real input)
Defines the frequency position of the RF spectrum in the input signal: normal or inverse. The position is important for correct down-conversion and filtering.

- "Normal" position of the RF spectrum



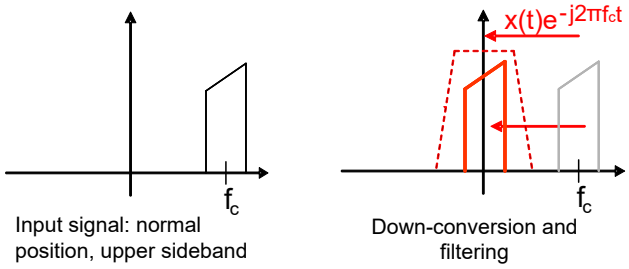
- "Inverse" position of the RF spectrum



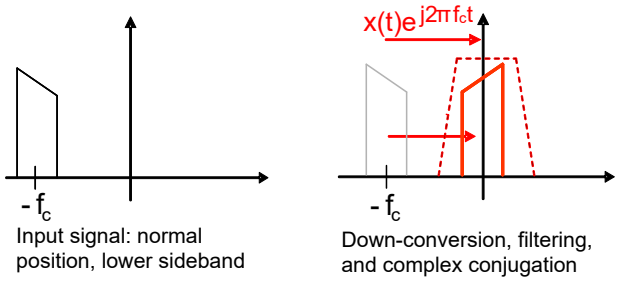
Remote command:
`CHANnel<m>:IQ:SBRF` on page 2475

Sideband (complex IF input)
Defines the sideband and the frequency position of complex modulated input signal in IF range.

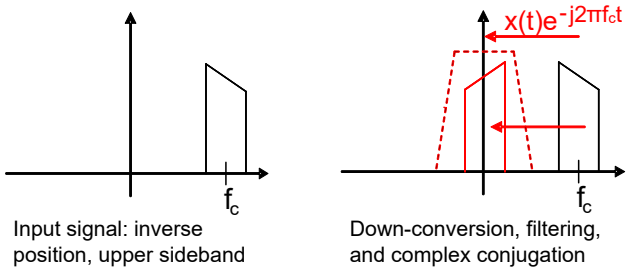
- "Normal/Upper": Upper sideband in normal position



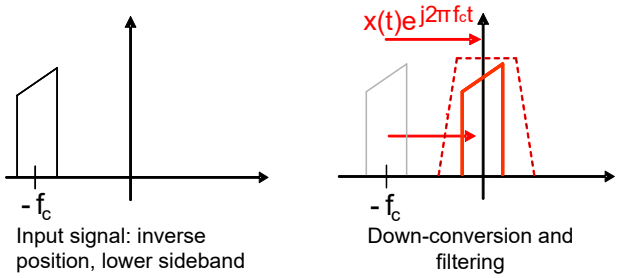
- "Normal/Lower": Lower sideband in normal position



- "Inverse/Upper": Upper sideband in inverse position



- "Inverse/Lower": Lower sideband in inverse position



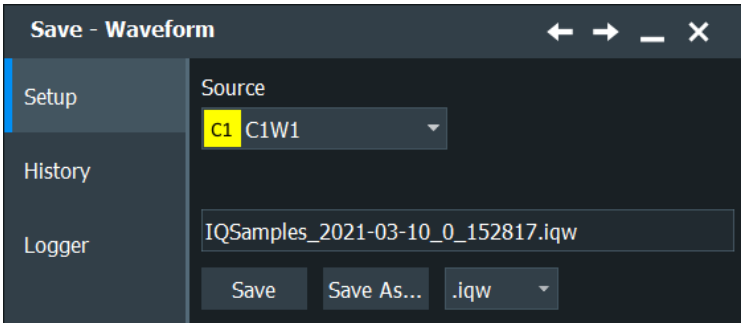
Remote command:
`CHANnel<m>:IQ:SBIF` on page 2476

17.3 I/Q data output

The recorded I/Q data can be saved manually on hard disc or USB flash device for further analysis, or extracted using remote control from an external computer. If supported

by the Rohde & Schwarz analysis software, a direct connection and data usage via LAN is also possible.

Access: "Menu" > "Save/Recall" > "Save" tab > "Waveform"



In I/Q mode, the "Save Waveform" dialog provides all functions to save the recorded I/Q data.

Source
Selects the channel for which the I/Q data is saved. For each input channel, a separate output file is written.

In case of a complex input signal that requires two input channels, the results of sources C1W1 and C2W1 are identical, as well as the results of C3W1 and C4W1.

Save IQ data to file
Enter the file name to save the I/Q data to. Double-tap the file name to open the file selection dialog box.

By default, the file name has the prefix "IQSamples_".

You can define a pattern for automatic naming in the "Autonaming" dialog, see Chapter 4.8.1, "Autonaming", on page 104.

"Save"	Saves the I/Q data in the selected file.
"Save As..."	Opens the file selection dialog box and saves the I/Q data to the selected file. See also Chapter 12.5, "File selection dialog", on page 482.
".iqw/.iq.tar/.bin/.xml/.csv"	Selects the file format. IQW: specific format for analysis with R&S I/Q data analysis software. It contains the I and Q values in interleaved order. IQ.TAR: iq-tar file format, packed format containing several files: an I/Q parameter XML file, an I/Q data binary file, and an optional I/Q preview XSLT file (style sheet). A detailed specification of the iq-tar file format is given in http://www.rohde-schwarz.com/en/manual/manual-r-s-iq-tar-file-format-specification-manuals-gb1_78701-37313.html. CSV, XML, and BIN: usual formats, which are also used for common waveform export, see Chapter 12.2.1, "Waveform export files", on page 453.

Remote command:
`CHANnel<m>:IQ:DATA[:VALues]?` on page 2477
`CHANnel<m>:IQ:DATA:HEADer?` on page 2477

17.4 NFC trigger

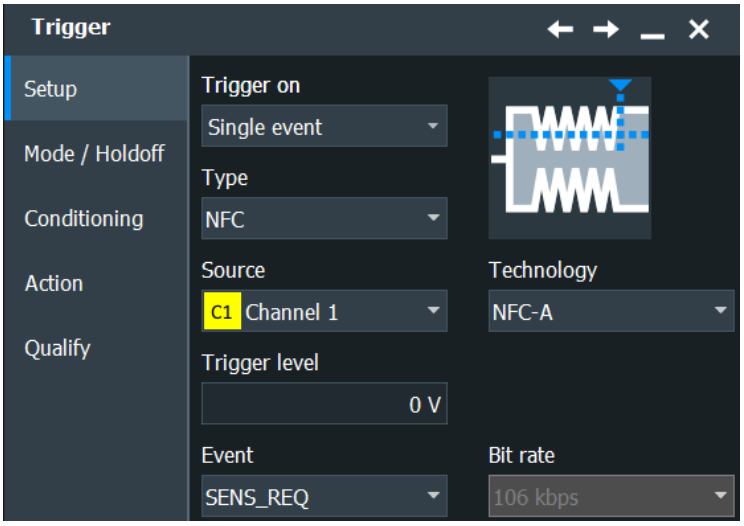
The Near Field Communication (NFC) trigger triggers on characteristic events of NFC signals, in particular on polling requests of a device, for example, of a mobile phone.

The trigger source is the NFC signal connected to one of the channel inputs.

You can analyze the triggered signal in two ways:

- In normal instrument mode, you can display and analyze the signal in the time domain.
- In IQ mode, you can acquire and save I/Q data for external analysis on a computer. The R&S RTO6 display shows the magnitude of the I/Q vector.

Access: "Menu">"Trigger" > "Setup" tab > "Type = NFC"



Technology
Selects the NFC technology, the communication protocol used by the input signal.

"NFC-A"	Compatible with ISO/IEC 14443A
"NFC-B"	Compatible with ISO/IEC 14443B
"NFC-F"	Compatible with JIS X6319-4 (FeliCa)

Remote command:
`TRIGger<m>:NFC:TECHnology` on page 2478

Bit rate
Sets the bit rate of the signal. For NFC-A and NFC-B, the bit rate is always 106 kBit per second. For NFC-F, the bit rate can be either 212 or 424 kBit per second.

Remote command:
`TRIGger<m>:NFC:BITRate` on page 2478

Event
Sets the trigger on:

- NFC-A:
  ALL_REQ or SENS_REQ command (polling requests)
- NFC-B:
  ALLB_REQ or SENSB_REQ command
- NFC-F:
  Start of sequence pattern with 48 bit or 96-bit length

Remote command:
`TRIGger<m>:NFC:EVENt` on page 2478

Trigger level
Sets the voltage level for the trigger condition. You can also drag the trigger level marker on the display (TA or TB on the right edge of the display). The range of the trigger level is limited in a way so that always a hysteresis for stable trigger conditions is available.

Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1391

18 Jitter analysis and clock data recovery

Jitter describes the timing errors in a system. It is a significant and undesired factor in high-speed serial communication designs because it causes transmission errors.

R&S RTO6 provides different options for analyzing jitter:

General jitter analysis (R&S RTO6-K12)

The jitter analysis option R&S RTO6-K12 provides common analysis and visualization tools for signal integrity analysis and jitter characterization:

- Automated jitter measurements in time domain like period, frequency, cycle-to-cycle jitter and skew
- Track graph of jitter measurement results
- Jitter spectrum
- Eye mask definition and analysis
- Software-based clock data recovery

See: Chapter 18.1, "Jitter measurements (options R&S RTO6-K12/K13)", on page 1069.

Advanced jitter analysis (R&S RTO6-K133)

The advanced jitter analysis option provides tools for detailed analysis including:

- All functionalities provided with option R&S RTO6-K12
- All data for total jitter, random jitter, periodic jitter, data-dependant jitter and deterministic jitter.
- Allows separation into horizontal and vertical components.
- Provides options for extensive result presentation, like histogram, spectrum, track, jitter bathtub graphics and synthetic eye diagram.

See: Chapter 18.2, "Advanced jitter and noise (option R&S RTO6- K133/K134)", on page 1079

Advanced jitter and noise analysis (R&S RTO6-K134)

The advanced noise analysis provides tools for detailed jitter and noise analysis including:

- All functionalities provided with option R&S RTO6-K133
- All data for total noise, random noise, periodic noise, data-dependant noise and deterministic noise.
- Allows separation into horizontal and vertical components.
- Provides options for extensive result presentation, like histogram, spectrum, track, noise bathtub graphics and synthetic eye diagram.

The different ways for analyzing jitter and noise are described in the following chapters.

In addition, the option R&S RTO6-K13 provides hardware-based clock data recovery. The resulting clock edge stream can be used as trigger source for the CDR trigger.

See: Chapter 18.2, "Advanced jitter and noise (option R&S RTO6- K133/K134)", on page 1079.

- Jitter measurements (options R&S RTO6-K12/K13).....1069
- Advanced jitter and noise (option R&S RTO6- K133/K134).....1079
- Clock data recovery.....1112

18.1 Jitter measurements (options R&S RTO6-K12/K13)

This chapter describes the measurement setup in the "Jitter" category.

- Jitter measurements.....1069
- Jitter measurement settings.....1071
- Jitter statistics and histogram.....1077
- Track of jitter measurement results.....1078
- Jitter spectrum.....1079

18.1.1 Jitter measurements

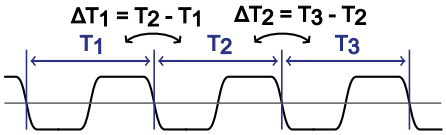
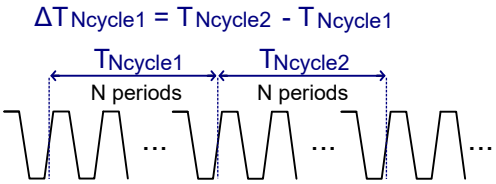
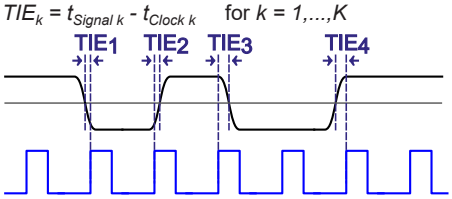
The measurement category "Jitter" gathers all measurements which are useful for jitter analysis. The category contains jitter-specific measurements. In addition, the following amplitude/time measurements are added to the jitter category, because they are useful for jitter analysis:

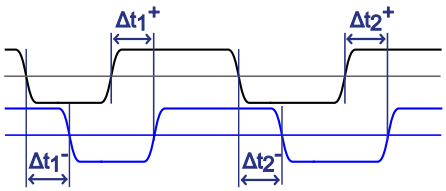
- Period
- Frequency
- Setup time
- Hold time
- Setup/Hold ratio

The amplitude/time measurements are described in "Time measurements" on page 318.

The specific jitter measurements are described in Table 18-1.

Table 18-1: Jitter measurements

	Measurement	Description/Result
	Cycle-cycle jitter	Difference between the periods of two adjacent cycles. The measurement is based on the period measurement. You can select the slope and the reference level on which the period is measured. $\Delta T_{Period\ k} = T_{Period\ k+1} - T_{Period\ k}$ for $k = 1,...,K-1$  See also: Chapter 18.1.2.2, "Settings for clock measurements", on page 1072
	N-cycle jitter	Difference between the time of two adjacent groups of N cycles (periods) each. You can select the slope and the reference level on which the time is measured, and the number of periods in the groups. $\Delta T_{Ncycle\ k} = T_{Ncycle\ k+1} - T_{Ncycle\ k}$ for k = *cycle group index* 
	Cycle-cycle width	Difference between the pulse width of two adjacent cycles. The measurement is based on the pulse width measurement. You can select the pulse polarity to be measured. $\Delta T_{Pulse\ k} = T_{Pulse\ k+1} - T_{Pulse\ k}$ for $k = 1,...,K-1$
	Cycle-cycle duty cycle	Difference between the duty cycle of two adjacent cycles. The measurement is based on the duty cycle measurement. You can select the pulse polarity for the duty cycle measurement. $\Delta R_{Cyc\ k} = R_{Cyc\ k+1} - R_{Cyc\ k}$ for $k = 1,...,K-1$
	Time interval error	Time difference between the slope of the input signal and the slope of a reference signal. The reference signal can be a captured clock waveform, or a clock generated by clock data recovery (CDR, software algorithm or hardware generation). You can select the slope and the reference level on which the TIE is measured. $TIE_k = t_{Signal\ k} - t_{Clock\ k}$ for $k = 1,...,K$  See also: Chapter 18.1.2.3, "Data measurement settings", on page 1074
	Unit interval	Period of the clock signal. If no clock signal is available, it is recovered by CDR. The period is calculated as the time difference between two consecutive clock edges of the same polarity. $UI_k = t_{Clock\ k+1} - t_{Clock\ k}$ for $k = 1,...,K-1$

	Measurement	Description/Result
	Data rate	Frequency of the clock signal. If no clock signal is available, it is recovered by CDR. The measurement is based on the unit interval measurement. $R_{Clock\ k} = 1 / UI_k$ for $k = 1,...,K-1$
	Skew delay	Delay between the edges of two interdependent waveforms. The measurement is a simplified variant of the "Delay" measurement assuming that both sources are similar except for the delay. $Skew\ delay = \Delta t_k = t_{Source2} - t_{Source1}$ for $k = 1,...,K$  See also: Chapter 18.1.2.4, "Delay measurement settings", on page 1076
	Skew phase	Phase difference between the edges of two waveforms. $Skew\ phase = Skew\ delay / Period * 360° = \Delta t_k / \Delta T_{Period\ k} * 360°$

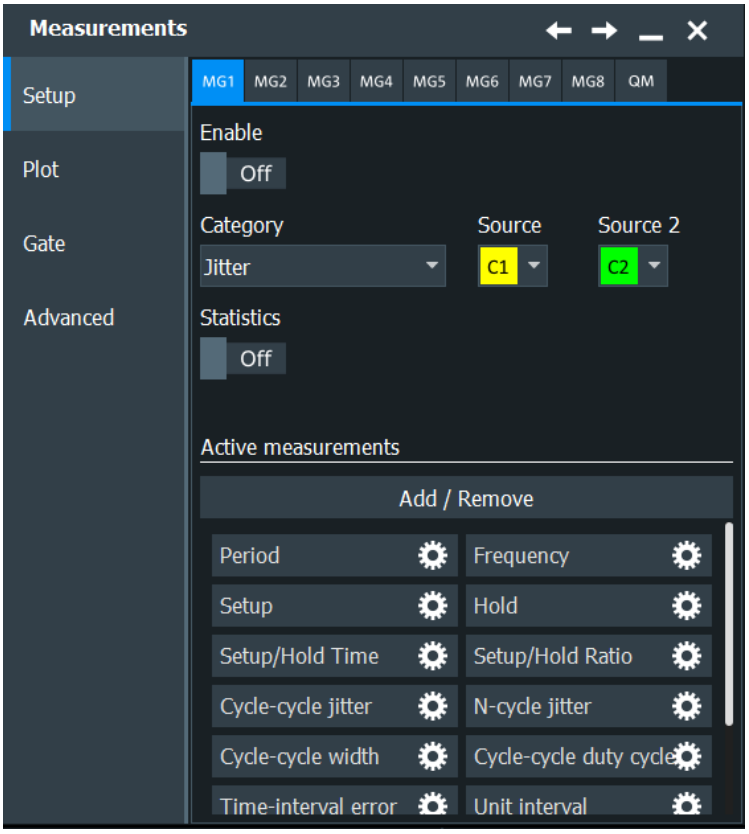
Limit and margin checks are also available for jitter measurements, see Chapter 8.2.13, "Limit and margin checks", on page 363. Limit and margin checks are based on the amplitude/time measurements.

18.1.2 Jitter measurement settings

18.1.2.1 Measurement selection

Access: [Measure] > "Setup" tab > "Jitter" category

Jitter measurements are only available for sources in the time domain.



Active Measurements

The "Active Measurements" list shows the measurements that are selected in the selected category. At least, one measurement must be selected. If further settings are available for a measurement, a settings icon is shown beside the measurement's name.

To change the selection, tap "Add / Remove".

For details on the available measurements, see Chapter 18.1.1, "Jitter measurements", on page 1069.

Remote command:

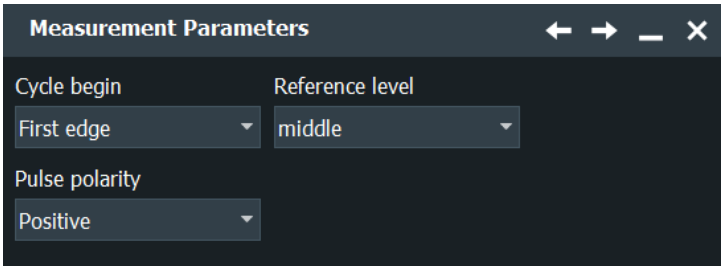
`MEASurement<m>:MAIN` on page 1483

`MEASurement<m>:ADDitional` on page 1484

18.1.2.2 Settings for clock measurements

The cycle-cycle measurements are intended to analyze the quality of clock signals. They require a few additional settings for period and pulse width measurement. See also: Table 18-1

► To configure a measurement, tap it in the "Active Measurements" list.



Cycle begin
Selects the slope at which the periods and thus the jitter is measured.

The setting is available for the following measurements: cycle-cycle jitter, N-cycle jitter, and cycle-cycle duty cycle.

"First edge" Measures the period from the first edge that is found, no matter of its direction.

"Positive" Measures the period at positive going edges.

"Negative" Measures the period at negative going edges.

"Either" Measures the period at both positive and negative going edges. This option is useful, for example, to check the clock stability of a double data rate clock.

Remote command:
`MEASurement<m>:JITTer:CCSLope` on page 2480

Ref level
The setting defines the "Data ref level" for setup and hold measurements, and for time-interval error measurements (option R&S RTO6-K12). It selects the reference level of the data signal on which the time is measured. The intersection of slope and reference level defines the time point for measurements.

For clock jitter measurements (option R&S RTO6-K12), it sets the "Reference level" for the time measurement.

Remote command:
`MEASurement<m>:AMPTime:DATA<n>:LSELect` on page 1497

Pulse polarity
Sets the polarity of pulses for which the pulse width is measured to obtain the cycle-cycle width and the cycle-cycle duty cycle.

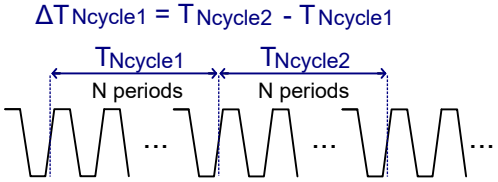
The setting is available for the following measurements: cycle-cycle width and cycle-cycle duty cycle.

"Positive" Pulse width of positive pulses is measured.

"Negative" Pulse width of negative pulses is measured.

Remote command:
`MEASurement<m>:JITTer:PULSe` on page 2481

Number of cycles
Sets the number of periods (cycles) that are accumulated to measure the N-cycle jitter.



Remote command:
`MEASurement<m>:JITTer:NCYCles` on page 2481

18.1.2.3 Data measurement settings

The following measurements are intended to analyze serial data. The clock can be a captured clock signal, or it can be recovered from the data signal using one of the clock recovery algorithms:

- Time interval error
- Unit interval
- Data rate

► To configure a measurement, tap it in the "Active Measurements" list.

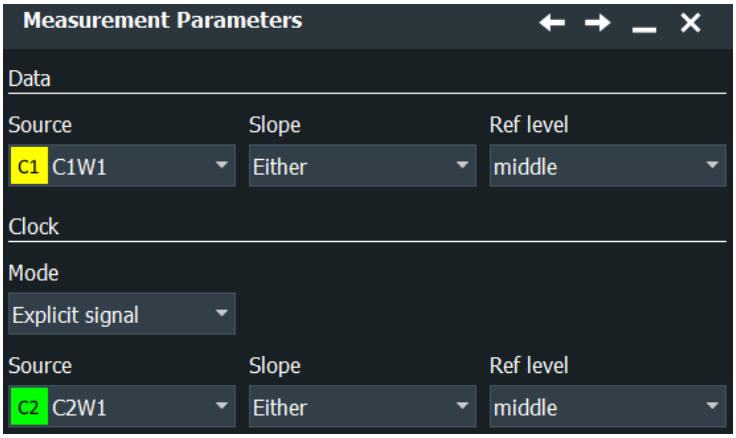


Figure 18-1: Settings for time interval error measurement with real clock signal

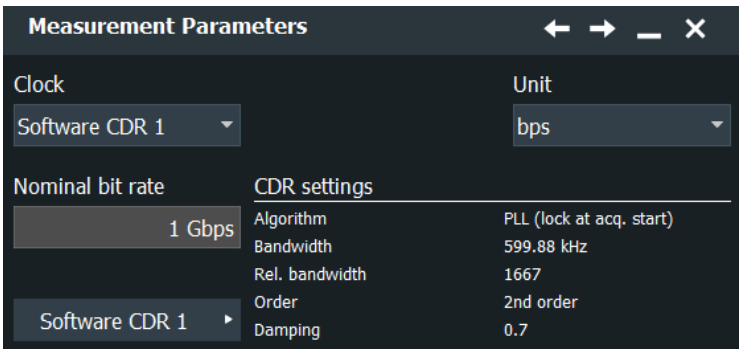


Figure 18-2: Settings for data rate measurement with software clock recovery

Clock
Defines the origin of the clock signal - either an existing clock signal or waveform, or a clock generated using one of the CDR methods.

The setting is available for the following measurements: time-interval error, unit interval and data rate.

"Explicit signal" The clock is an existing clock signal. Select the clock source, clock slope, and the reference level.

"Software CDR 1 / 2" The clock is generated by a software algorithm. The R&S RTO6 provides two setups for software CDR, which can be configured and used independently. To configure the CDR, tap the "Software CDR 1/2 setup" button.

"Hardware CDR" The clock is generated from the source signal by the integrated CDR circuitry. To configure the CDR, tap the "Hardware CDR setup" button.
The HW CDR must be active to perform the measurement. To activate HW CDR, use the trigger type "CDR", or the trigger type "Serial pattern" with active CDR.

Note: For TIE measurement results with HW CDR, set "UI offset = 0". Other values introduce a timing offset in the results.

Remote command:
`MEASurement<m>:JITTer:CDRMode` on page 2482

Slope
Sets the edges that are used for measurements.

For time-interval error measurement, you set the data slope, and the clock slope if the clock is a real signal. For unit interval and data rate measurements, you set only the clock slope if the clock is a real signal.

"Positive" The positive clock slope can be used, for example, for single data rate (SDR) signals with bit start at the positive clock edge.

"Negative" The negative clock slope can be used, for example, for SDR signals with bit start at the negative clock edge.

"Either" For clock edges, this option can be used for double data rate (DDR) signals.
For data edges, it is the most common setting.

Remote command:
`MEASurement<m>:JITTer:SOURce<n>:TIESlope` on page 2482

Clock ref level
Selects the reference level of the clock on which the time is measured. The intersection of slope and reference level defines the time point for measurements.

The setting is used for setup and hold measurements, and for jitter measurements (option R&S RTO6-K12).

Remote command:
`MEASurement<m>:AMPTime:CLCK<n>:LSELect` on page 1497

Ref level
The setting defines the "Data ref level" for setup and hold measurements, and for time-interval error measurements (option R&S RTO6-K12). It selects the reference level of the data signal on which the time is measured. The intersection of slope and reference level defines the time point for measurements.

For clock jitter measurements (option R&S RTO6-K12), it sets the "Reference level" for the time measurement.

Remote command:
`MEASurement<m>:AMPTime:DATA<n>:LSELect` on page 1497

Unit
Sets the unit to bps (bit per second), or to Hz for analysis of radar signals. The setting is available for data rate measurements.

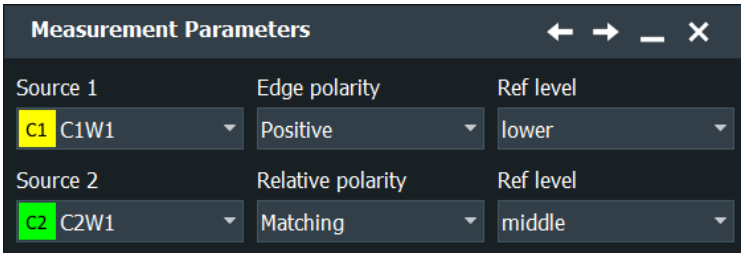
Remote command:
`MEASurement<m>:JITTer:DRUNit` on page 2483

18.1.2.4 Delay measurement settings

Skew delay and skew phase measurements are intended to measure the time difference between the edges of two waveforms. The measurements are simplified variants of the "Delay" and "Phase" measurements assuming that both sources are similar except for the delay.

See also: Table 18-1

► To configure a measurement, tap it in the "Active Measurements" list.



Source 1, Source 2
"Source 1" is the reference signal, and "Source 2" is the signal compared to the reference signal.

Skew delay = $\Delta t_k = t_{Source2} - t_{Source1}$ for $k = 1,\ldots,K$

Edge polarity
Sets the edge of the first waveform from which the skew delay or phase is measured: positive, negative or both.

Remote command:
`MEASurement<m>:JITTer:SKWSlope` on page 2483

Relative polarity
Sets the edge of the second waveform relative to the first waveform.

"Matching" Measures from positive to positive edge or from negative to negative edge.

"Inverse" Measures from positive to negative edge or from negative to positive edge.

Remote command:
`MEASurement<m>:JITTer:SKWRelation` on page 2484

Data ref level / Clock ref level
See "Ref level" on page 326 and "Clock ref level" on page 325.

18.1.2.5 Example: measuring TIE with HW CDR

To measure the time interval error with hardware-recovered clock, the option R&S RTO6--K13 is required.

1. Configure the HW CDR settings according to the characteristics of the measured signal.
2. Set up the trigger:
   a) "Menu">"Trigger" > "Setup" > "Source" = channel of the input signal > Trigger type = CDR
   b) Tap "Find level" to determine the trigger level.
3. Configure the measurement:
   a) On the "Measure" menu, select "Setup".
   b) Select the "Jitter" category.
   c) Tap "Add / Remove" and select "Time interval error".
   d) Tap "Time interval error" in the "Active measurements" list.
   e) Select the "Data slope" and the reference level to be used to measure the jitter.
   f) Select the "Clock = Hardware CDR".
   g) In the "Measure" > "Setup" tab, enable the measurement.
   h) Enable "Statistics".

18.1.3 Jitter statistics and histogram

Since jitter is a random component of all signals, statistical measurement results are required to characterize the jitter.

To get measurements statistics of a jitter measurement

Prerequisite: the jitter measurements are selected and configured, and the measurement group is active.

1. Double tap the "Meas Results" box.

   The "Measurement" toolbar assist opens.

2. In the overlay menu, enable "Statistics state".
3. Tap "Advanced Setup".
4. On the "Measure" > "Advanced" tab, enable "Measure all events in each acquisition".

The histogram plots the density of data. It shows the frequency of occurrence of the measurement values. The maximum count of a measurement value is assigned to the full height of the histogram diagram (= 1000). All other count values are displayed relative to the maximum.

To enable the histogram

1. On the "Plot" tab, enable the "Histogram".
2. If the histogram is not displayed as expected, disable "Continuous auto scale" and adjust the "Meas scale".

18.1.4 Track of jitter measurement results

A track graph displays the results of the first jitter measurement from a single acquisition as a time-correlated waveform. To generate the track graph, multiple measurement points are required. Thus, enabling the track automatically activates "Measure all events in each acqu.".

You can measure amplitude and time on the track waveform: Configure a new measurement group, e.g. "MG2", that uses the track waveform as measurement source.

You can also zoom into the track waveform, perform cursor measurements on it, and export the track.

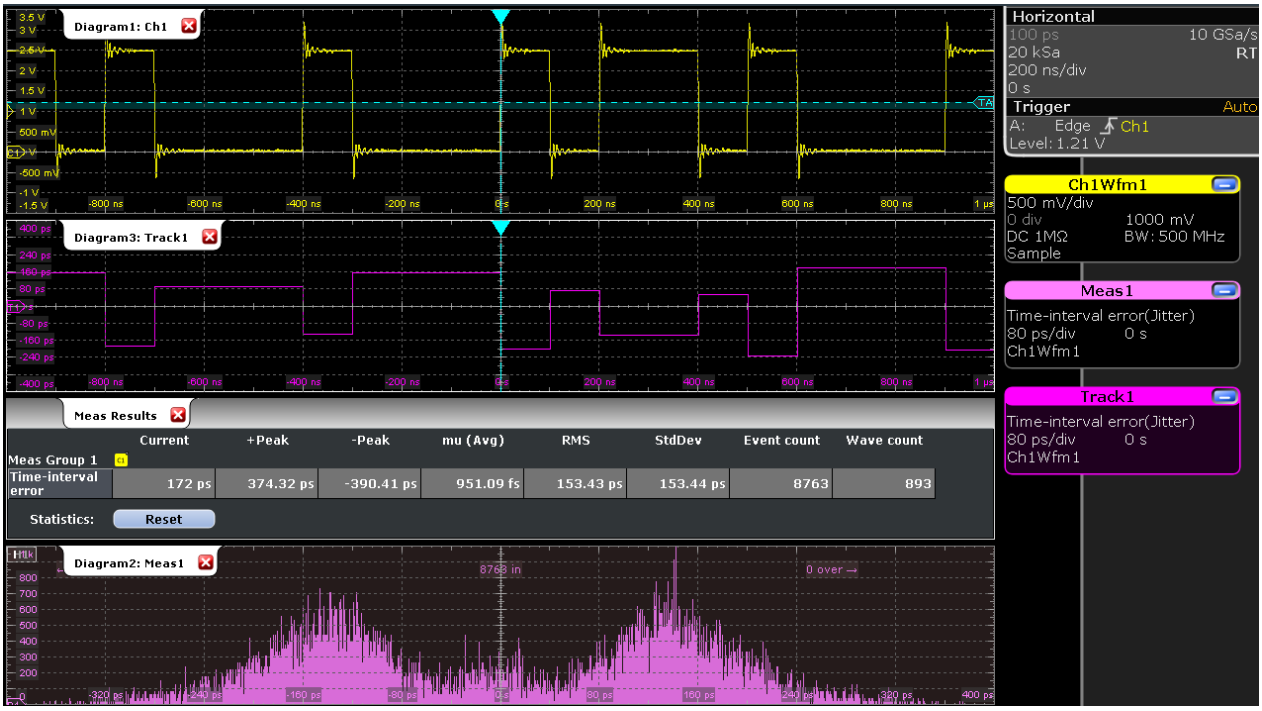


Figure 18-3: TIE measurement with CDR trigger, 10 Mbps, with track and histogram

18.1.5 Jitter spectrum

An extension of jitter measurements is the FFT analysis of track waveforms of jitter measurements. The results are displayed in the frequency domain as a jitter spectrum. The modulation frequency is displayed on the horizontal axis, and the amplitude of modulation on the vertical axis. Using FFT, periodic components can be detected.

18.2 Advanced jitter and noise (option R&S RTO6- K133/K134)

This chapter describes the settings of the advanced jitter option R&S RTO6- K133 and of the advanced noise option R&S RTO6- K134.

18.2.1 About jitter

Jitter typically consists of various jitter components that are caused by different jitter sources, see Figure 18-4.

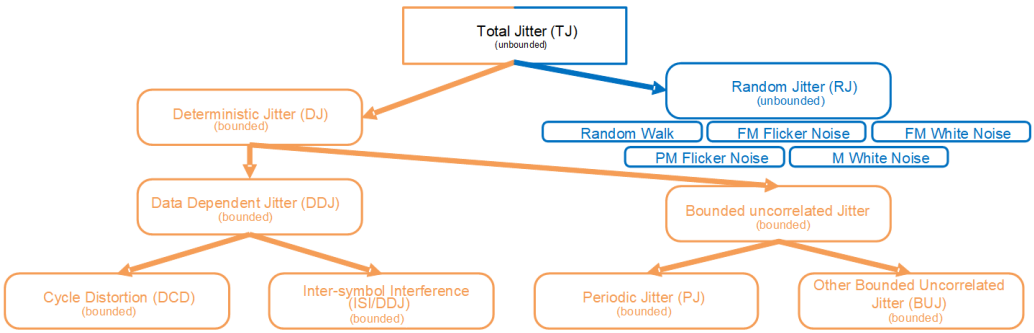


Figure 18-4: Jitter decomposition tree

With the advanced jitter option R&S RTO6- K133, you can perform different measurements for calculating and displaying the different jitter components.

For the analysis of jitter, it is important to understand these sources and the contributions to the total jitter (TJ). For an analytical approach, a jitter model is frequently used, which splits jitter into the two major categories of random and deterministic jitter.

The jitter of the different components has different causes. Knowing what jitter component is influencing your system can help you identify what phenomenon causes the jitter.

Deterministic jitter

Deterministic jitter (DJ), also called systematic jitter, is further broken down into periodic jitter (PJ) and data-dependent jitter (DDJ). Deterministic jitter is bounded and specified as peak-to-peak value.

Periodic jitter

Periodic jitter (PJ) is caused by a periodic disturbance. Though this signal is not necessarily sinusoidal, it is frequently also named sinusoidal jitter. The amplitude of the periodic signal bounds the jitter.

A strong local RF oscillator, a switch-mode power supply, undesired crosstalk or an unstable, oscillating PLL cause period jitter due to an unintentional coupling into the signal.

Data-Dependent jitter

Inter-symbol interference (ISI) causes data-dependent jitter (DDJ). When ISI is present, the signal is disturbed with an attenuated, time-shifted copy of itself or spectral parts of itself.

Random jitter

Random jitter is unbounded and commonly specified by the standard deviation σ. Due to its irregular nature, random jitter (RJ) is uncorrelated to any other signal and unpredictable in timing behavior. Contributions to the random jitter are thermal noise, shot noise, 1/f noise and other physical effects.

18.2.2 About noise

Noise describes the amplitude deviation of a data symbol at the sampling time respective to a corresponding reference level. These deviations or level errors (LE) are caused by various sources.

The total noise is split into two main components: random noise (RN) and deterministic noise (DN). DN consists of periodic noise (PN) and non-periodic components.

The advanced noise algorithm enables various selectable result plots of the activated noise components in the results tab.

18.2.3 Configuring the advanced jitter

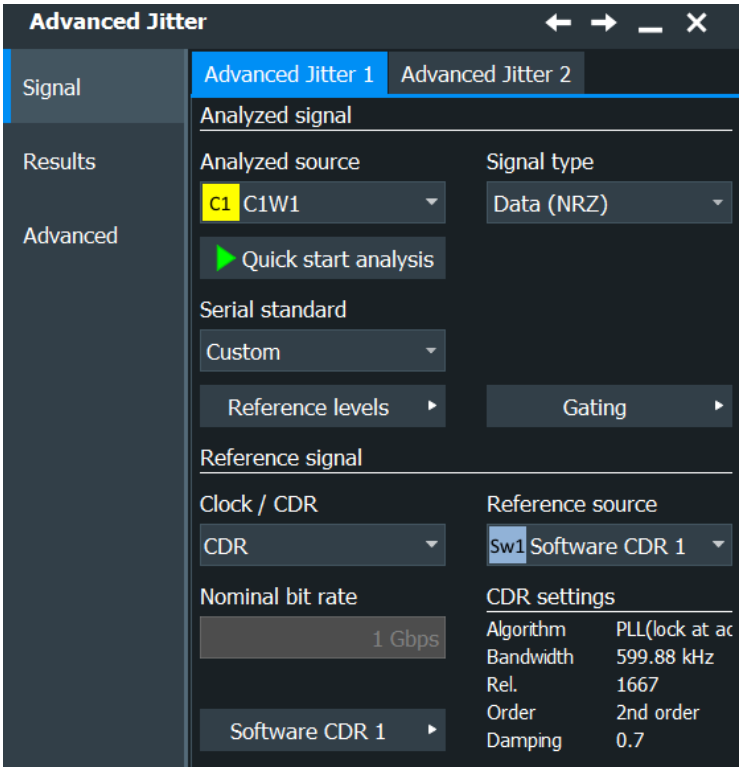
To perform an advanced jitter analysis:

1. Open the "Analysis" > "Advanced Jitter" menu.
2. In the "Signal" tab, set up the analyzed signal and reference signal.
3. In the "Advanced Settings" tab, configure the "Data dependent" settings .
4. If necessary, enable "Periodic"/"(Other) bounded uncorrelated".
5. In the "Result" tab, enable all general and component-specific results diagrams, that you want to display.
6. In the "Result" tab, "Add/ remove components". A dialog opens, where you can enable all required measurements.
7. Press "Enable" to view all selected measurement results and diagrams.

18.2.4 Signal settings

Access: App Cockpit> "Analysis" > "Advanced Jitter / Noise" > "Signal" tab

In the "Signal" tab, you can set up the signal you want to analyze and define the reference clock signal.



Analyzed source
Selects the source for the advanced jitter analysis.

Remote command:
`ADVJitter<m>:SIGNal:SOURce` on page 2486

Serial standard
For some serial standards, the bit rate is known to the instrument. If you select the standard, the nominal bit rate for the advanced jitter analysis is set automatically.

The serial standards are available only if the instrument is able to support its maximum nominal bit rate.

Serial standard	Nom. bit rate	Bandwidth	Rel. bandwidth	Order	Damping
USB 2.0	480 Mbps	1667	287.947 kHz	1st	-
USB 3.1 Gen 1	5 Gbps	500	10 MHz	2nd	0.7
Displayport 1.1	1.296 Gbps	1667	777.44 kHz	1st	-
Displayport 1.2	4.32 Gbps	1667	2.59 MHz	1st	-
Fibre Channel 1x	1 Gbps	1667	599.88 kHz	1st	-
Fibre Channel 2x	2 Gbps	1667	1.2 MHz	1st	-

Serial standard	Nom. bit rate	Bandwidth	Rel. bandwidth	Order	Damping
Fibre Channel 4x	4 Gbps	1667	2.4 MHz	1st	-
HDMI 1.2	1.32 Gbps	1667	791.84 kHz	1st	-
HDMI 1.4	3.4 Gbps	1667	2.04 MHz	1st	-
HDMI 2.0	4.8 Gbps	1667	2.88 MHz	1st	-
PCIe Gen 1	2.5 Gbps	1667	1.5 MHz	2nd	0.7
PCIe Gen 2	5 Gbps	1000	5 MHz	2nd	0.7
SATA Gen 1 (1/500)	1.5 Gbps	500	3 MHz	2nd	0.7
SATA Gen 1 (1/1667)	1.5 Gbps	1667	899.82 kHz	2nd	0.7
SATA Gen 1	1.5 Gbps	1667	899.82 kHz	2nd	0.7
SATA Gen 2 (1/500)	3 Gbps	500	6 MHz	2nd	0.7
SATA Gen 2 (1/1667)	3 Gbps	1667	1.8 MHz	2nd	0.7
SATA Gen 2	3 Gbps	1667	1.8 MHz	2nd	0.7
SATA Gen 3 (1/500)	6 Gbps	500	12 MHz	2nd	0.7
SATA Gen 3 (1/1667)	6 Gbps	1667	3.6 MHz	2nd	0.7
SATA Gen 3	6 Gbps	1667	3.6 MHz	2nd	0.7
SAS Gen 1	3 Gbps	1667	1.8 MHz	1st	-
SAS Gen 2	6 Gbps	1667	3.6 MHz	1st	-
XAUI	3.75 Gbps	1667	2.25 MHz	1st	-

Remote command:
`ADVJitter<m>:SIGNal:SERStandard` on page 2491

Signal Type
Select, if the analyzed signal is a data or clock signal.

Remote command:
`ADVJitter<m>:SIGNal:TYPE` on page 2485

Quick start analysis
Starts a quick advanced jitter analysis.

Before starting the quick start analysis, set the Analyzed source and the Signal Type .

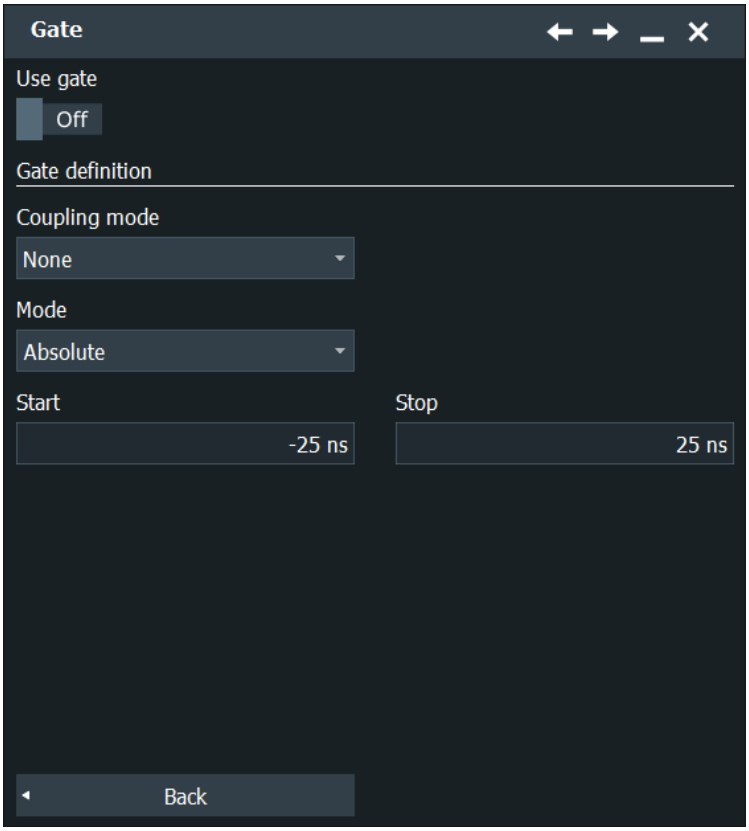
The quick start analysis performs the following:

- A bit rate estimation is performed on the applied signal.
- A software CDR of second order is used as reference signal for the jitter analysis.
- Enables basic measurement "Event Count" and advanced jitter components: TJ@BER, RJ, TJ, DJ and PJ.
- In the "Result" dialog it enables:
  – "Bathtub" for total jitter @ BER (TJ@BER).
  – "Step response" for the analyzed source signal.
  – "Histogram" for RJ, TJ, DJ, DDJ and PJ.
- Enables the advance jitter measurement analysis.

Remote command:
`ADVJitter<m>:SIGNal:QUICkmeas` on page 2486

Reference Level
Opens a dialog to set the reference levels. For details, see Chapter 8.2.4, "Reference levels", on page 307.

Gating
Opens a dialog to set up the gate of the waveform for the advanced jitter analysis. The gate limits the measurement to a user-defined range of the waveform.



Use gate ← Gating
Enables the gating setup.

Remote command:
ADVJitter<m>:SIGNal:GATE:STATe on page 2491

Coupling mode ← Gating
Selects if the gate is coupled to the zoom or cursor.

To use the coupling mode, you have to enable a cursor or zoom first.

Remote command:
ADVJitter<m>:SIGNal:GATE:COUPling on page 2489

Cursor / Zoom ← Gating
Selects the cursor/ zoom the gate is coupled to, if "Coupling mode" is set to "Cursor "/"Zoom".

Remote command:
ADVJitter<m>:SIGNal:GATE:CURSor on page 2489
ADVJitter<m>:SIGNal:GATE:ZDIagram on page 2491

Mode ← Gating
Selects if the gate area is defined in relative values (percent) or absolute.

Remote command:
ADVJitter<m>:SIGNal:GATE:RANGe:MODE on page 2490

Start / Stop ← Gating
Sets the start/stop time of the gate area.

Remote command:
ADVJitter<m>:SIGNal:GATE:RANGe:ABSolute:STARt on page 2489
ADVJitter<m>:SIGNal:GATE:RANGe:ABSolute:STOP on page 2490

Relative start / stop ← Gating
Sets the relative start/stop of the gate area in percent.

Remote command:
ADVJitter<m>:SIGNal:GATE:RANGe:RELative:STARt on page 2490
ADVJitter<m>:SIGNal:GATE:RANGe:RELative:STOP on page 2490

Clock / CDR
Selects if a clock or a CDR signal is used as a reference signal.

"CDR"	Uses software clock data recovery.
"Clock"	Uses an explicit clock signal. The signal has to be provided on an additional channel.

Remote command:
ADVJitter<m>:SIGNal:REFSource on page 2486

Reference source
Selects the reference source for the clock/CDR signal.

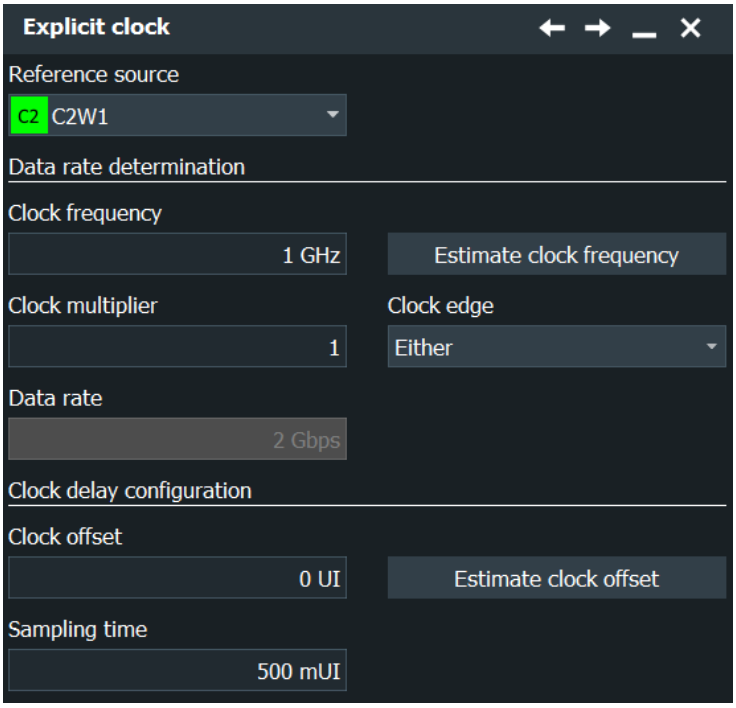
Remote command:
ADVJitter<m>:SIGNal:CLKSource on page 2487
ADVJitter<m>:SIGNal:CDRMode on page 2486

Nominal bit rate
Sets /displays the nominal bit rate. It corresponds to the data rate of the data stream from which the clock is to be recovered.

Remote command:
`ADVJitter<m>:SIGNal:BITRate` on page 2486

Software CDR Setup
Opens a dialog for configuring the CDR. For details, see Chapter 18.3, "Clock data recovery", on page 1112.

Explicit clock
Opens a dialog to define the setting of the explicit clock.



Clock frequency ← Explicit clock
Sets the clock freuqnecy of the explicit clock signal. If this value is unknown, you can use "Estimate clock frequency".

Remote command:
`ADVJitter<m>:SIGNal:CLFRequency` on page 2488

Estimate clock frequency ← Explicit clock
Calculates an estimate value for the clock frequency.

Remote command:
`ADVJitter<m>:SIGNal:CLEStimation` on page 2487

Clock multiplier ← Explicit clock
Sets a value for the clock multiplier, the ratio of an internal clock rate to the externally supplied clock.

Remote command:
`ADVJitter<m>:SIGNal:CLFFactor` on page 2487

Clock edge ← Explicit clock
The clock edge sets the point in time when the state of the data signal is checked:

"Positive" Rising edge, a positive voltage change.

"Negative" Falling edge, a negative voltage change.

"Either" Both the rising and the falling edge.

Remote command:
`ADVJitter<m>:SIGNal:EDGE` on page 2487

Data rate ← Explicit clock
Displays the calculated data rate. It is calculated as follow:

*"Data rate" = "Clock frequency" * "Clock multiplier"*X*

Where *X* is:

- 1 for "Clock edge"= "Positive" or "Negative"
- 2 for "Clock edge"= "Either"

Remote command:
`ADVJitter<m>:SIGNal:DATRate?` on page 2488

Clock offset ← Explicit clock
Sets a clock offset. If this value is unknown, you can use "Estimate clock offset".

Remote command:
`ADVJitter<m>:SIGNal:CLOFfset` on page 2488

Estimate clock offset ← Explicit clock
Calculates an estimate value for the clock offest.

Remote command:
`ADVJitter<m>:SIGNal:CFEStimation` on page 2488

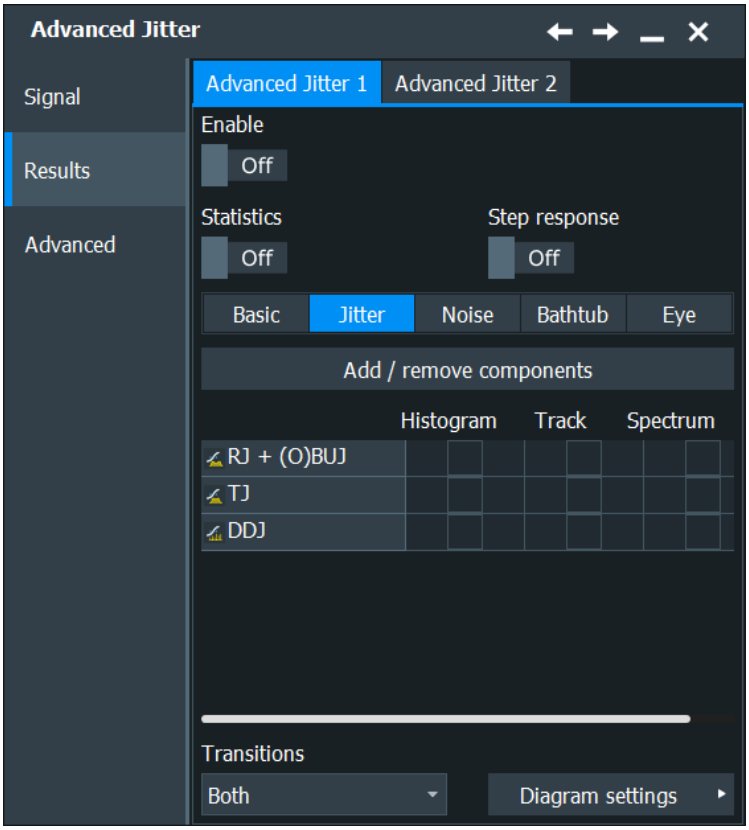
Sampling time ← Explicit clock
Sets a sampling time for the explicit clock signal, an offset for the clock edge in relation to the bit start. The clock edge sets the beginning of the unit interval. The sampling time is a number between 0 and 1. Value 0 sets the clock edge to the beginning of the bit period; value 0.5 sets the clock edge to the middle of the bit period.

Remote command:
`ADVJitter<m>:SIGNal:SAMTime` on page 2489

18.2.5 Results

Access: "Analysis" menu > "Advanced Jitter" > "Result" tab.

In this tab, you can select the jitter components measurements that you want to perform and enable the results.



The advanced jitter option provides different ways to display your measurement results. In the "Result" tab, you can select which measurement results you want to display.

Figure 18-5 and Figure 18-6 show an example of the displayed advanced jitter and noise analysis results.

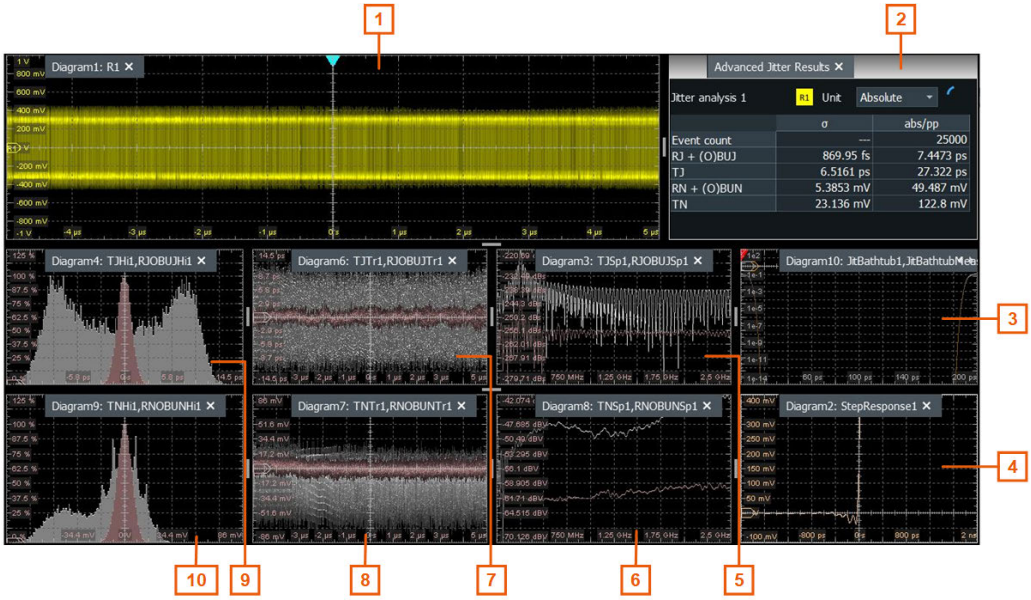


Figure 18-5: Advanced jitter and noise analysis results display

1 = Analyzed signal spectrum
2 = Advanced jitter results
3 = Jitter bathtub
4 = Noise bathtub
5 = Jitter spectrum
6 = Noise spectrum
7 = Jitter track
8 = Noise track
9 = Jitter histogram
10 = Noise histogram

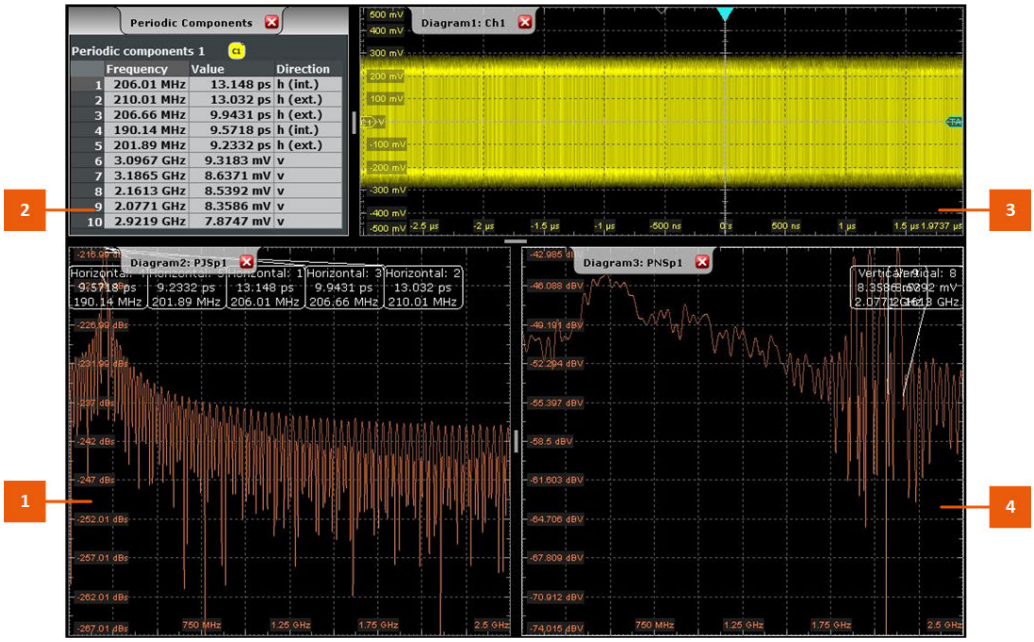


Figure 18-6: Advanced jitter and noise periodic results display

1 = Periodic jitter spectrum
2 = Periodic components results
3 = Analyzed signal spectrum
4 = Periodic noise spectrum

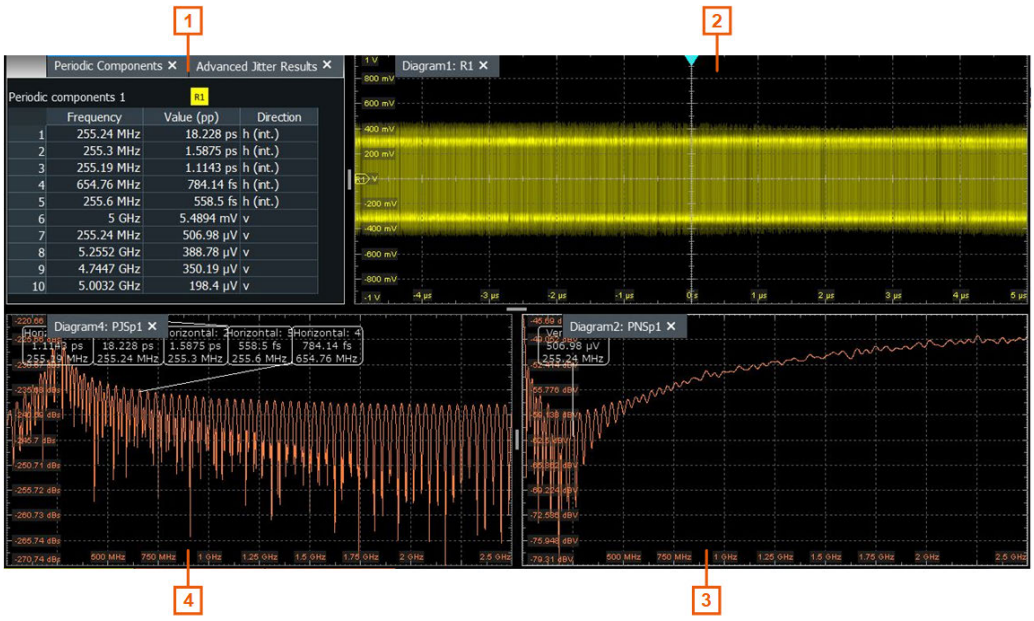


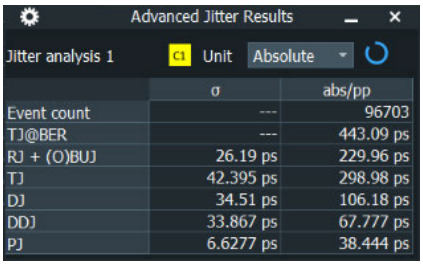
Figure 18-7: Advanced jitter and noise periodic results display

1 = Periodic components results
2 = Analyzed signal spectrum
4 = Periodic jitter spectrum
3 = Periodic noise spectrum

18.2.5.1 General settings

In the "General Settings" section of the "Results" dialog, you can enable the advanced jitter analysis, and select which results you want to display.

Enable
Enables the advanced jitter analysis. It displays all enabled diagrams and displays a table with the results for the component measurements. It also displays the measurement results for all enabled Basic components , Jitter Components.



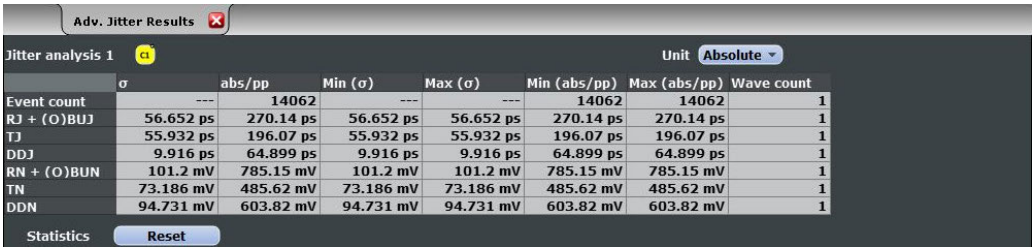
Remote command:
`ADVJitter<m>:RESult:ENABle` on page 2494

Statistics

Enables the calculation and display of statistical results. They include:

- "Min (abs/pp)": absolute minimum value of the component.
- "Max (abs /pp)": absolute maximum value of the component.
- "Min (σ)": minimum value of the standard deviation.
- "Max (σ)": maximum value of the standard deviation.
- "Wave count": Number of waveforms (acquisitions) the measurement is based on.

In the "Advanced Jitter Results" table, you can reset the statistics.



	σ	abs/pp	Min (σ)	Max (σ)	Min (abs/pp)	Max (abs/pp)	Wave count
Event count	---	14062	---	---	14062	14062	1
RJ + (O)BUJ	56.652 ps	270.14 ps	56.652 ps	56.652 ps	270.14 ps	270.14 ps	1
TJ	55.932 ps	196.07 ps	55.932 ps	55.932 ps	196.07 ps	196.07 ps	1
DDJ	9.916 ps	64.899 ps	9.916 ps	9.916 ps	64.899 ps	64.899 ps	1
RN + (O)BUN	101.2 mV	785.15 mV	101.2 mV	101.2 mV	785.15 mV	785.15 mV	1
TN	73.186 mV	485.62 mV	73.186 mV	73.186 mV	485.62 mV	485.62 mV	1
DDN	94.731 mV	603.82 mV	94.731 mV	94.731 mV	603.82 mV	603.82 mV	1

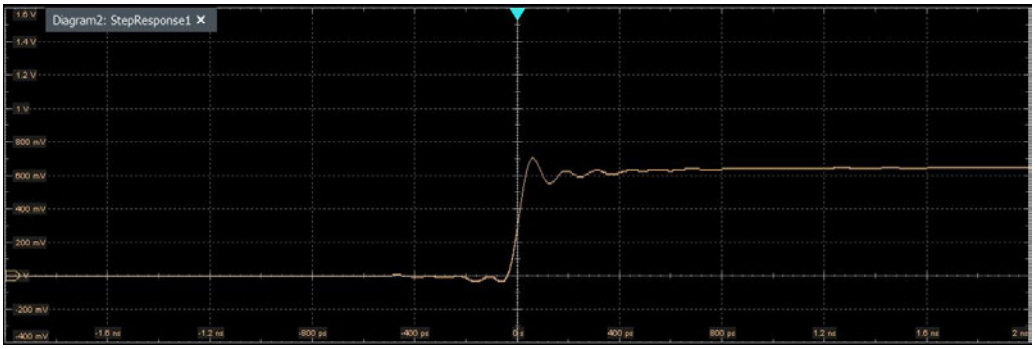
Remote command:
`ADVJitter<m>:RESult:STATistics` on page 2495

Step response

Enables a step response diagram. The diagram displays the time behavior of the system when the inputs change from zero to one or from one to zero within a short time period.

You can set the Length and the Step position for the step response diagram in the "Advanced Settings" dialog.

For further analysis of the step response, you can use it as a source for "Amp/Time" measurements. See Chapter 8.2.5, "Amplitude/time measurements", on page 316.

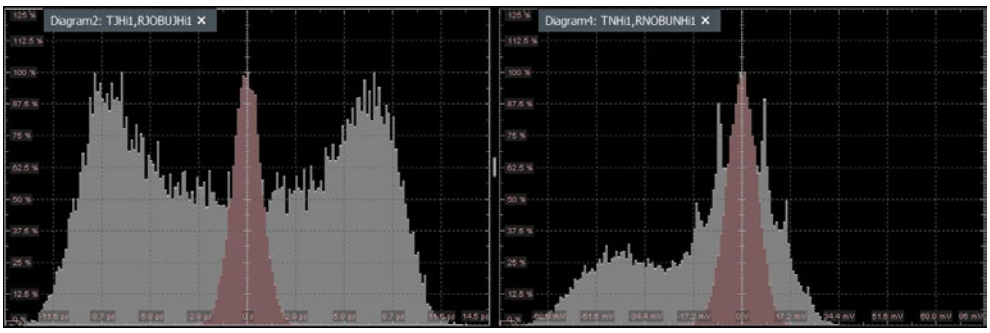


Remote command:
`ADVJitter<m>:RESult:STEPresponse` on page 2495

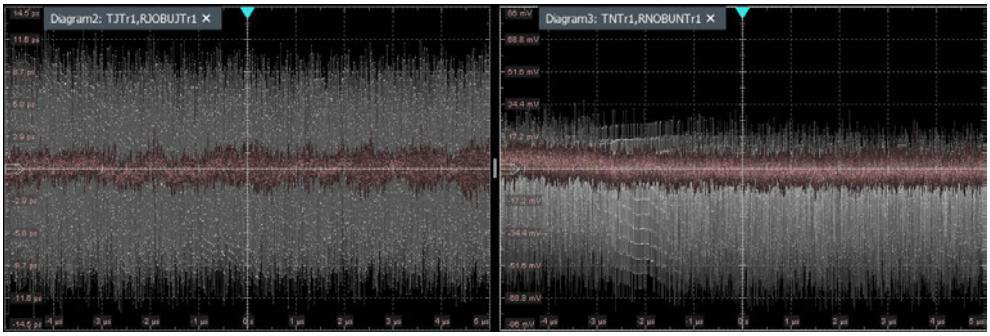
Component specific results

For the selected jitter component, selected you can display:

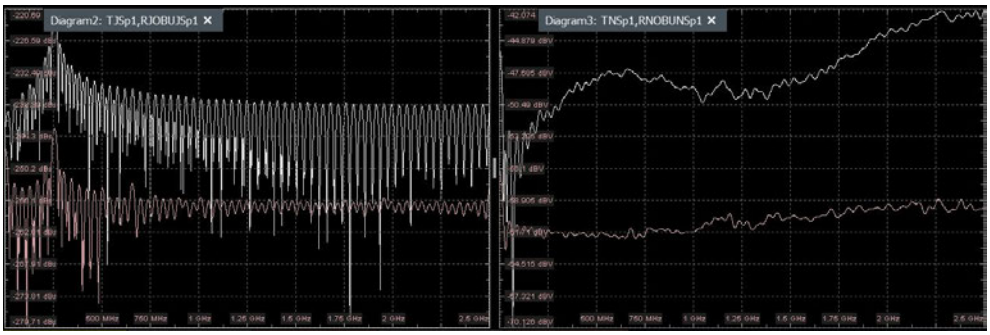
- "Histogram": displays a histogram of measurement results for the selected component - the cumulative occurrence distribution of measurement results in a graphic. You can enable a cursor and use it for further analysis.



- "Track": the measurement results over time for an acquired waveform. It is the graphical interpretation of all measurement values of a single acquisition. Unlike the histogram, it reveals trends of change in the analysis and preserves the timing relationship of the measurement results to the signal.



- "Spectrum": displays how the power of the signal is distributed over the frequency.



Remote command:
`ADVJitter<m>:RESult:COMPonents<n>:HISTogram` on page 2495
`ADVJitter<m>:RESult:COMPonents<n>:TRACk` on page 2496
`ADVJitter<m>:RESult:COMPonents<n>:SPECtrum` on page 2496

Unit

In the jitter results dialog, you can select the unit for the displayed results. The following values are available:

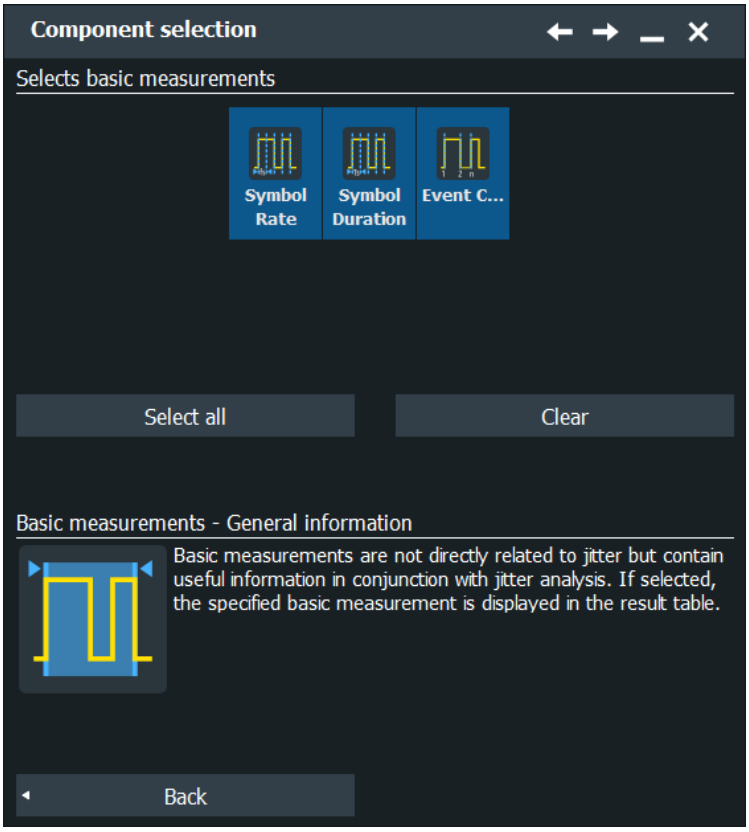
- For jitter measurements, you can select between absolute (seconds) and relative (unit intervals) values.
- For noise measurements, you can select between absolute (V) and relative (%) values.

Remote command:
`ADVJitter<m>:RESult:UNIT` on page 2496

18.2.5.2 Basic measurement settings

Access: "Analysis" menu > "Advanced Jitter" > "Result" tab > "Basic".

In the "Basic" tab, you can add measurements for better understanding of the collected data. You can then enable a histogram or track on them, see Component specific results.



Add / remove components
Opens a dialog to select the jitter and noise components.

Basic components
To display the measurement results, "Enable" the advanced jitter analysis in the "Results" tab.

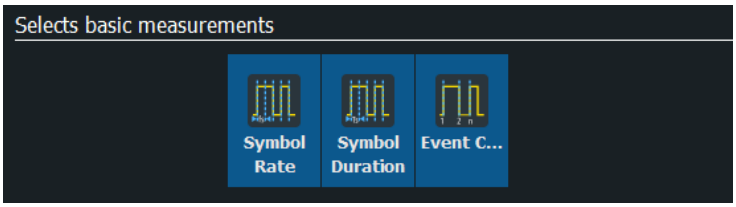


Table 18-2: Advanced jitter basic measurements

Measurement	Abbreviation	Description	StdDev	p/abs	Histogram	Track
Symbol rate	f_symb	Number of data symbols per second of the analyzed signal. It is equivalent to the baud rate. For binary signals, the signal rate is equal to the bit rate.		X	X	X
Symbol duration	T_symb	The length of time to transfer one signal symbol. Equals the inverse value of the symbol rate.		X	X	X
Event Count	EC	Number of analyzed data transitions.		X		

Remote command:
`ADVJitter<m>:DCOMposition:COMPonents<n>:ENABle` on page 2495

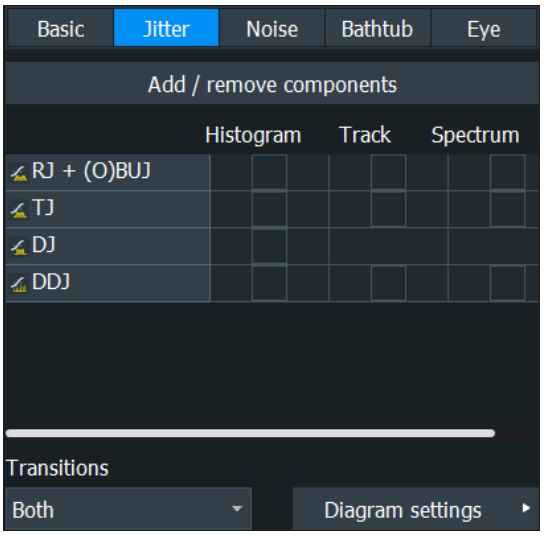
Select all /Clear
Selects/clears all components in the basic dialog.

Remote command:
`ADVJitter<m>:DCOMposition:SELBasicmeas` on page 2499
`ADVJitter<m>:DCOMposition:CLEBasicmeas` on page 2499

18.2.5.3 Jitter measurement settings

Access: "Analysis" menu > "Advanced Jitter" > "Result" tab > "Jitter".

In the "Jitter" tab, you can add the jitter components you want to measure. You can then enable a histogram, track or spectrum on them, see Component specific results.



Add / remove components
Opens a dialog to select the jitter components.

Jitter Components
In the "Jitter" tab, you can select which jitter components you want to measure. For the enabled measurements you can display the results in different forms, see Chapter 18.2.5.1, "General settings", on page 1090.

Table 18-3: Advanced jitter components measurements

Measurement	Abbreviation	Description	Results		Diagram		
			StdDev	ap/abs	Histogram	Track	Spectrum
Total jitter at BER	TJ@BER	The eye closure at the user-defined target BER.		x			
Total jitter	TJ	Measures the total Time Interval Error (TIE) of the input signal. It is defined as the time between each transition of the input signal and its corresponding reference clock transition.	x	x	x	x	x
Deterministic jitter	DJ	The reproducible portion of the overall TIE of the input signal. It consists of the Data Dependent Jitter (DDJ), Periodic Jitter (PJ) and the Other Bounded Uncorrelated Jitter (OBUJ) components.		x			
Data-dependent jitter	DDJ	Consists of Inter-symbol Interference (ISI) and Duty Cycle Distortion (DCD). It depends on the symbol sequence. ISI is an effect commonly caused by limited signal bandwidth in the transmitter and transmission line, which forces influence from past symbols onto future symbols. DCD is a mismatch between rising and falling signal transitions due to reference level mismatch or other imbalances between transitions.	x	x	x	x	x
Duty cycle distortion(3)	DCD	Measure of imbalance between rising and falling data transitions. It is defined as the time difference between the average rising and average falling edge.		x			
Random plus (other) bounded uncorrelated jitter	RJ + (O)BUJ	Random jitter plus bounded uncorrelated jitter/ other bounded uncorrelated jitter is the combined effect of both jitter types. If periodic components are activated, RJ + OBUJ is displayed. Otherwise RJ + BUJ. RJ + (O)BUJ is typically used, if track or spectrum plots are desired. Both plots are not available for RJ only or (O)sBUJ only components as the time dependencies of these jitter types are unknown. RJ + (O)BUJ spectrum is a useful approximation of the RJ spectrum for negligible small (O)BUJ.	x	x	x	x	x

(1) Available, if (Other) bounded uncorrelated > "Enabled".

(2) Available, if Periodic > "Enabled".

(3) No results available, if "Transitions" is set to "Rising"/"Falling".

Measurement	Abbreviation	Description	Results		Diagram		
			StdDev	p/abs	Histogram	Track	Spectrum
Periodic jitter(2)	PJ	Caused by crosstalk from periodic signal sources to the transmitter or transmission line. It represents the harmonic portion of the overall time interval error (TIE).	x	x	x	x	x
Deterministic Dual-Dirac jitter	DJ (δ-δ)	Industry standard approximation to the deterministic jitter measurement, calculated by fitting a model of two overlapping, equally large Gaussian distributions to the observed TJ histogram. The calculated value is the time difference between the two distributions center points.		x			
Inter-Symbol interference	ISI	The time between the earliest and the latest signal transition of a single transition type, i.e.rising and falling edges. The measurement result is the largest value across all transition types.		x			
Data-dependent and periodic jitter(2)	DDJ + PJ	Consist of the data-dependent and periodic jitter DDJ+PJ is the same as deterministic jitter minus other bounded uncorrelated jitter. In contrast to deterministic jitter, DDJ+PJ offers track and spectrum plots. DDJ+PJ spectrum is a useful approximation of the deterministic jitter spectrum for negligible small other bounded uncorrelated jitter.	x	x	x	x	x
Other bounded uncorrelated jitter(1)	(O)BUJ	Composite impact of all unknown but deterministic portions of the overall time interval error of the input signal. It consists of any disturbance that comes from a deterministic source but cannot be reproduced from the input signal. Often it is caused by crosstalk from other data signals.	x	x	x		

(1) Available, if (Other) bounded uncorrelated > "Enabled".

(2) Available, if Periodic > "Enabled".

(3) No results available, if "Transitions" is set to "Rising"/"Falling".

Measurement	Abbreviation	Description	Results		Diagram		
			StdDev	ap/abs	Histogram	Track	Spectrum
Other bounded uncorrelated Dual-Dirac jitter(1)	(O)BUJ (δ -δ)	Other bounded uncorrelated Dual-Dirac jitter (O)BUJ (δ- δ) is an industry standard approximation to the other bounded uncorrelated jitter measurement. It is calculated by fitting a model of two overlapping, equally large Gaussian distributions to the observed RJ+(O)BUJ histogram. The calculated value is the amplitude difference between the two distributions center points.		x			
Random jitter (1)	RJ	Consists of any influence that cannot be predicted with model parameters. Usually, this jitter component follows a Gaussian probability distribution. The algorithm computes the standard deviation and the peak-peak value of the TIE caused by RJ within the analyzed waveform.	x		x		

(1) Available, if (Other) bounded uncorrelated > "Enabled".

(2) Available, if Periodic > "Enabled".

(3) No results available, if "Transitions" is set to "Rising"/"Falling".

To display the measurement results, "Enable" the advanced jitter analysis in the "Results" tab.

Remote command:
`ADVJitter<m>:DCOMposition:COMPonents<n>:ENABle` on page 2495

Select all /Clear
Selects/clears all components in the jitter dialog.

Remote command:
`ADVJitter<m>:DCOMposition:CLEJitcomp` on page 2499
`ADVJitter<m>:DCOMposition:SELJitcomp` on page 2499

Transitions
Selects which transitions are considered for the display of the diagrams, rising, falling or both. This setting is mostly relevant for histograms, tracks and the measurement results.

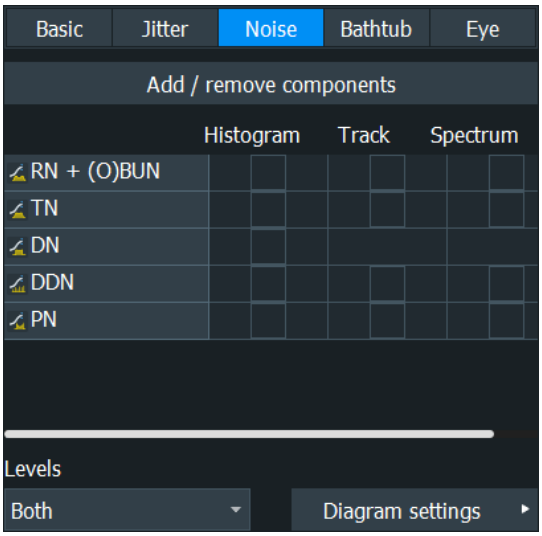
Remote command:
`ADVJitter<m>:RESult:TRANsitions` on page 2500

18.2.5.4 Noise measurement settings

Available for option R&S RTO6- K134.

Access: "Analysis" menu > "Advanced Jitter" > "Result" tab > "Noise".

In the "Noise" tab, you can add the noise components you want to measure. You can then enable a histogram, track or spectrum on them, see Component specific results.



Add / remove components
Opens a dialog to select the noise components.

Noise components
In the "Noise" tab, you can select which jitter noise components you want to measure.

Component selection
Select noise components
EH@BER
RN + (O)BUN
TN
DN
DDN
LD
PN
ISIN
DDN + PN
(O)BUN
(O)BUN (δ-δ)
RN
Select all
Clear
Enable (Other) bounded uncorrelated in tab "Advanced"
Noise - General information
Noise describes the amplitude deviation of a data symbol at the sampling time with respect to a corresponding reference level. These deviations or Level Errors (LE) are caused by various sources.
The Total Noise (TN) is split into two main components: Random Noise (RN) and Deterministic Noise (DN). DN consists of Periodic Noise (PN) and non periodic components.
Back

Measurement	Abbreviation	Description	Results		Diagram		
			StDev	ap/abs	Histogram	Track	Spectrum
Eye height @BER[3]	EH@BER	Returns the vertical eye opening at the user-defined BER.		x			
Random plus (other) bounded uncorrelated noise	RN+ (O)BUN	Random noise plus (other) bounded uncorrelated noise or (other) bounded uncorrelated noise is the combined effect of both noise types. If periodic components are activated RN + OBUN are displayed, otherwise RN + BUN. RN + (O)BUN is typically used, if track or spectrum plots are desired. Both plots are not available for RN only or (O)BUN only components as the time dependencies of these noise types are unknown. RN + (O)BUN spectrum is a useful approximation of the RN spectrum for negligible small (O)BUN.	x	x	x	x	x
Total noise	TN	Measures the total level error of the input signal. It is defined as the amplitude deviation between the amplitude of the input signal at the decision time point and its corresponding reference level.	x	x	x	x	x
Deterministic noise	DN	The reproducible portion of the overall level error of the input signal. It consists of the data-dependent noise, the periodic noise components and the other bounded uncorrelated jitter components.	x	x	x		
Data-dependent noise	DDN	Consists of inter-symbol interface noise and level distortion. It depends on the symbol sequence. ISIN is an effect whereby past symbols have an influence on future symbols. DCD is a mismatch between rising and falling signal transitions due to reference level mismatch or other imbalances between transitions.	x	x	x	x	x
Level distortion[3]	LD	A measure of imbalance between different symbol levels. It is defined as the difference between the average level errors of the symbols one and zero.		x			
Periodic noise[2]	PN	Periodic noise is typically caused by crosstalk from periodic signal sources to the transmitter or transmission line. It represents the harmonic portion of the overall level error.	x	x	x	x	x

[1] Available, if (Other) bounded uncorrelated > "Enabled".

[2] Available, if Periodic > "Enabled".

[3] No results available, if "Levels" is set to "Rising"/"Falling".

Measurement	Abbreviation	Description	Results		Diagram		
			StDev	ap/abs	Histogram	Track	Spectrum
Inter-symbol interference noise	ISIN	The level difference between the highest and the lowest level error of a symbol, i.e. one or zero.The largest value across all symbols is the measurement result.		x			
Data-dependent noise and periodic noise[2]	DDN+PN	Data-dependent noise plus periodic noise is the combined effect of both jitter types. Note that DDN+PN equals deterministic noise minus other bounded uncorrelated noise. DDN+PN spectrum is a useful approximation of the deterministic noise spectrum fot negligible small other bounded uncorrelated noise.	x	x	x	x	x
Other bounded uncorrelated noise [1]	(O)BUN	Composite impact of all unknown but deterministic portions of the overall time interval error of the input signal. It consists of any disturbance that comes from a deterministic source but cannot be reproduced from the input signal. Often it is caused by crosstalk from other data signals.	x	x	x		
Other bounded uncorrelated Dual-Dirac noise [1]	(O)BUN (δ-δ)	Other bounded uncorrelated Dual-Dirac noise (O)BUN (δ- δ) is an industry standard approximation to the other bounded uncorrelated jitter measurement. It is calculated by fitting a model of two overlapping, equally large Gaussian distributions to the observed RN+(O)BUN histogram. The calculated value is the amplitude difference between the two distributions center points.		x			
Random noise [1]	RN	Consists of any influence that cannot be predicted with model parameters. Usually, this noise component follows a Gaussian probability distribution. The algorithm computes the standard deviation and the peak-peak value of the level error (LE) caused by RN within the analyzed waveform.	x		x		

[1] Available, if (Other) bounded uncorrelated > "Enabled".

[2] Available, if Periodic > "Enabled".

[3] No results available, if "Levels" is set to "Rising"/"Falling".

Remote command:
`ADVJitter<m>:DCOMposition:COMPonents<n>:ENABle` on page 2495

Select all /Clear
Selects/clears all components in the noise dialog.

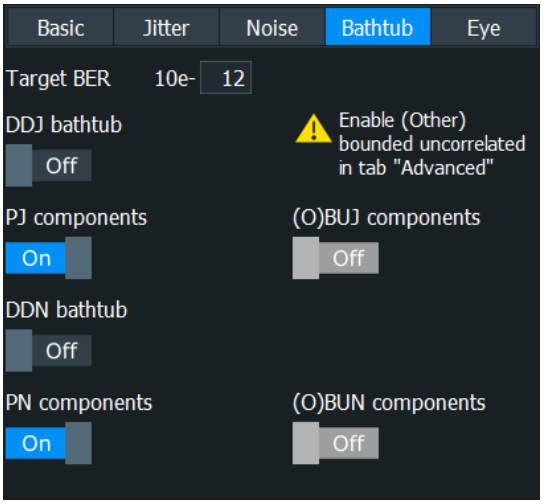
Remote command:
`ADVJitter<m>:DCOMposition:CLENoisecomp` on page 2499
`ADVJitter<m>:DCOMposition:SELNoisecomp` on page 2499

Levels
Selects which levels are considered for the display of the diagrams, rising, falling or both. This setting is mostly relevant for histograms, tracks and the measurement results.

Remote command:
`ADVJitter<m>:RESult:LEVels` on page 2500

18.2.5.5 Bathtub settings

Access: "Analysis" menu > "Advanced Jitter" > "Result" tab > "Bathtub".



Target BER 10
Sets the target bit error rate.

Remote command:
`ADVJitter<m>:DCOMposition:TARBer` on page 2500

DDJ bathtub
Enables a bathtub plot, a graph of BER in a log scale versus data-dependent jitter (DDJ).

Remote command:
`ADVJitter<m>:RESult:BATHtub` on page 2501

PJ / (O)BUJ components
Selects which jitter components are considered for the jitter bathtub.

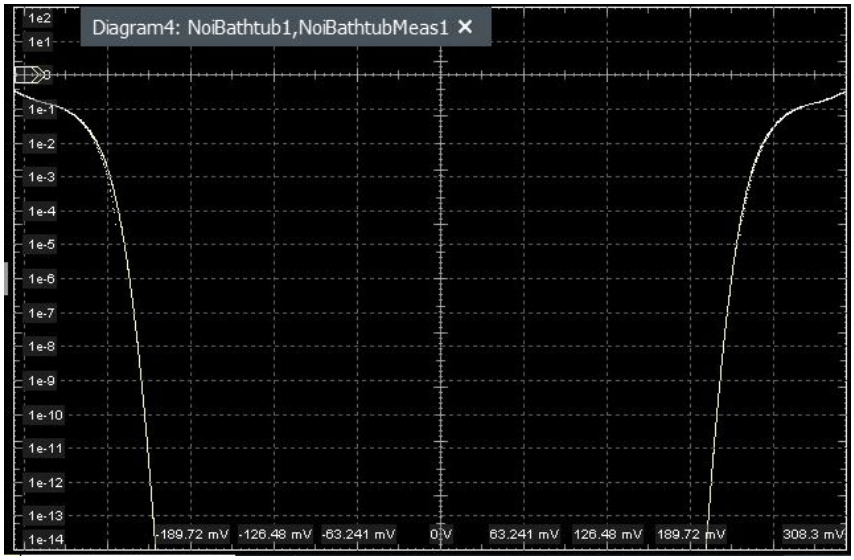
"PJ" is available, if Periodic > "Enabled".

"(O)BUJ"is available, if (Other) bounded uncorrelated > "Enabled".

Remote command:
`ADVJitter<m>:RESult:PJ` on page 2501
`ADVJitter<m>:RESult:OBUJ` on page 2501

DDN bathtub
Available for option R&S RTO6- K134.

Enables a noise bathtub plot, a graph of BER in a log scale versus data-dependent jitter (DDN).



Remote command:
`ADVJitter<m>:RESult:NBAThtub` on page 2502

PN / (O)BUN components
Selects which noise components are considered for the noise bathtub.

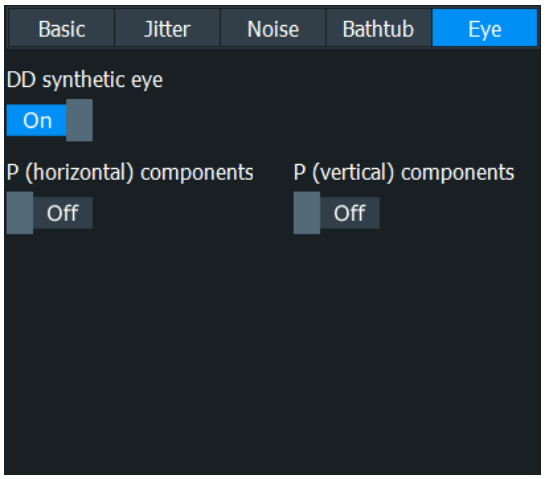
"PN" is available, if Periodic > "Enabled".

"(O)BUN" is available, if (Other) bounded uncorrelated > "Enabled".

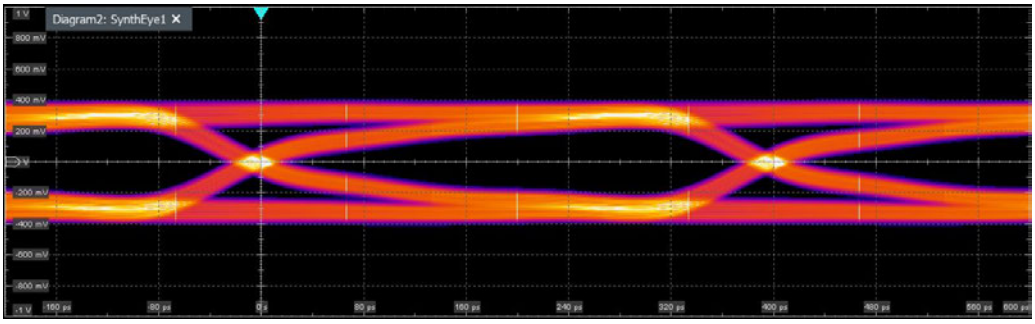
Remote command:
`ADVJitter<m>:RESult:PN` on page 2502
`ADVJitter<m>:RESult:OBUN` on page 2502

18.2.5.6 Eye settings

Access: "Analysis" menu > "Advanced Jitter" > "Result" tab > "Eye".



Synthetic eye
Enables a synthetic eye plot out of the synthesized data.



Remote command:
`ADVJitter<m>:RESult:SYNTheticeye` on page 2503

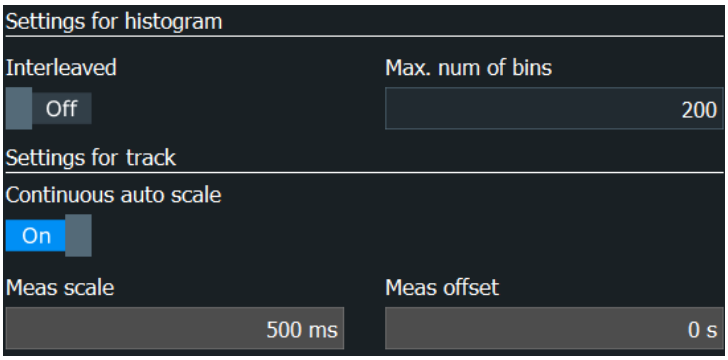
Add periodic components to DD synthetic eye
Adds horizontal/vertical periodic components to the data-dependent synthetic eye.

Remote command:
`ADVJitter<m>:RESult:PHORizontal` on page 2502
`ADVJitter<m>:RESult:PVERtical` on page 2503

18.2.5.7 Diagram settings

In the diagram settings dialog you can set a separate scale for the basic, jitter and noise parameters.

Settings for histogram and track
Defines the settings for the histogram and track diagrams.



Interleaved ← Settings for histogram and track
Displays interleaved histogram values.

Remote command:
`ADVJitter<m>:DCOMposition:JINTerleaved` on page 2507

Max. numb of bins ← Settings for histogram and track
Sets the number of bins, the number of data intervals, of the histogram.

Remote command:
`ADVJitter<m>:RESult:VERTical:HBINs` on page 2504

Continuous auto scale ← Settings for histogram and track
If enabled, the measurement scale for track is set automatically. If disabled you can manually set a Meas scale and Meas offset.

Remote command:
`ADVJitter<m>:RESult:VERTical:CONTinuous` on page 2504
`ADVJitter<m>:RESult:VERTical:BHContinuous` on page 2504
`ADVJitter<m>:RESult:VERTical:NHContinuous` on page 2505

Measurement type ← Settings for histogram and track
Selects the basic measurement type to apply the scale to.

Remote command:
`ADVJitter<m>:RESult:VERTical:BMTYpe` on page 2504

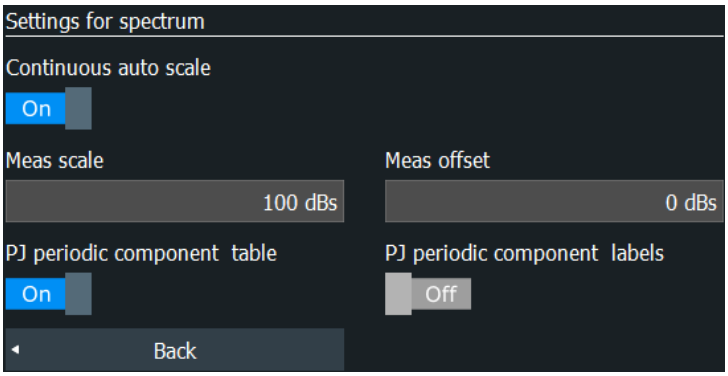
Meas scale ← Settings for histogram and track
Sets a measurement scale. The unit depends on the measurement type.

Remote command:
`ADVJitter<m>:RESult:COMPonents<n>:SCALe` on page 2505

Meas offset ← Settings for histogram and track
Sets a measurement offset. The unit depends on the measurement type.

Remote command:
`ADVJitter<m>:RESult:COMPonents<n>:OFFSet` on page 2506

Settings for spectrum
Defines the settings for the histogram and track diagrams.

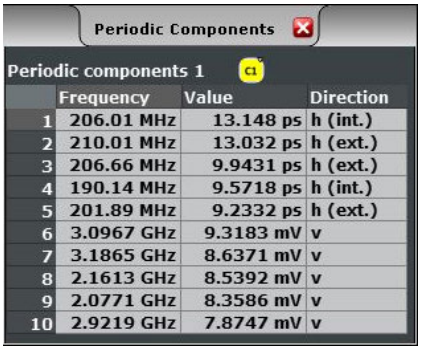


Continuous auto scale ← Settings for spectrum
If enabled, the measurement scale for the spectrum is set automatically. If disabled you can manually set a Meas scale and Meas offset.

Remote command:
`ADVJitter<m>:RESult:VERTical:JSContinuous` on page 2505
`ADVJitter<m>:RESult:VERTical:NSContinuous` on page 2505

PJ periodic component table ← Settings for spectrum
Enables the display of the periodic jitter components table. The table is displayed after the advanced jitter/ noise analysis is enabled.

The unit of the periodic components depends on the switching between "Absolute" and "Relative", that means times are represented in “s” and “UI” and voltages in “V” and “%”.



	Frequency	Value	Direction
1	206.01 MHz	13.148 ps	h (int.)
2	210.01 MHz	13.032 ps	h (ext.)
3	206.66 MHz	9.9431 ps	h (ext.)
4	190.14 MHz	9.5718 ps	h (int.)
5	201.89 MHz	9.2332 ps	h (ext.)
6	3.0967 GHz	9.3183 mV	v
7	3.1865 GHz	8.6371 mV	v
8	2.1613 GHz	8.5392 mV	v
9	2.0771 GHz	8.3586 mV	v
10	2.9219 GHz	7.8747 mV	v

Remote command:
`ADVJitter<m>:DCOMposition:SPJTable` on page 2506

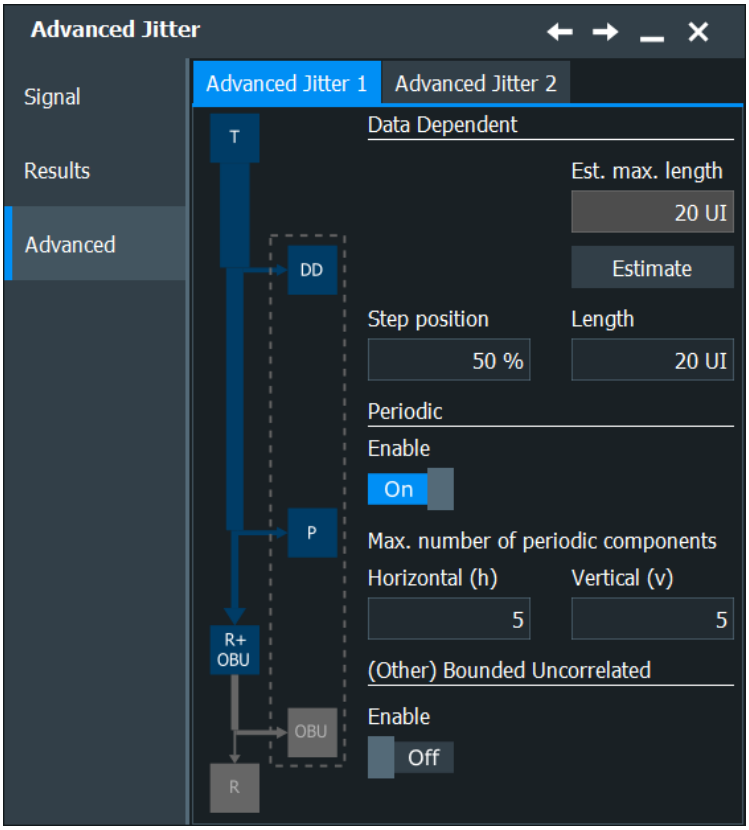
PJ periodic component labels ← Settings for spectrum
Displays the periodic jitter/noise labels on the spectrum diagrams.

Available, if in the "Result"> "Jitter" tab, "Spectrum" is enabled for the "Periodic Jitter" measurement.

Remote command:
`ADVJitter<m>:DCOMposition:SPJLabels` on page 2507
`ADVJitter<m>:DCOMposition:SPNLabels` on page 2507

18.2.6 Advanced settings

Access: "Analysis" menu > "Advanced Jitter" > "Advanced settings" tab.



Length
Sets the step response length in unit intervals. A higher step response value leads to longer measurement times. Hence it is best to set this value to the number of unit intervals that is roughly needed until the ringing has subsided.

You can set the step response manually or run an estimate.

The "Estimate" sets the maximum feasible value. If it is lower than the set "Length", then the "Length" value is overwritten.

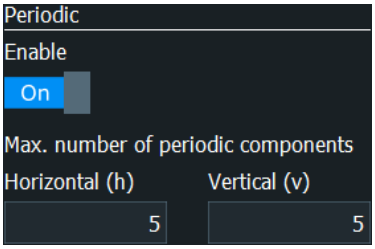
Remote command:
`ADVJitter<m>:DCOMposition:RESPlength` on page 2508
`ADVJitter<m>:DCOMposition:RLEStimate` on page 2508
`ADVJitter<m>:DCOMposition:MRESplength?` on page 2508

Step position
Sets the steps position. It defines what percentage of the unit lengths are displayed before the step response.

For example, if the "Length" is set to 20UI and the "Step position" is set to 75%, the step response diagram displays 15 UI before the step response and 5 UI after that.

Remote command:
`ADVJitter<m>:DCOMposition:POSStep` on page 2508

Periodic
If enabled, allows the selection of the "Periodic jitter" / "Periodic noise" components: PJ, DDJ+PJ, PN and DDN+PN.



Enable ← Periodic
Enables the periodic components.

Remote command:
`ADVJitter<m>:DCOMposition:ENP` on page 2509

Horizonta (h) direction ← Periodic
Sets the maximum number of periodic components in the horizontal direction that are considered for the measurement. If the actual number of periodic components is higher than the set value, they are considered for the random jitter component.

The horizontal components differentiate also between intrinsic and extrinsic. Extrinsic periodic horizontal components describe timing variations likely caused by external interference on the clock generation system of the DUT. The interference source is free-running and not coupled to the generated clock. Intrinsic periodic horizontal components describe timing variations likely caused by fluctuations within the clock generation system of the DUT. The interference is tightly coupled to the generated clock and follows modulations in data rate, e.g. spread spectrum clocking.

Remote command:
`ADVJitter<m>:DCOMposition:NHORizontal` on page 2509

Vertical (v) direction ← Periodic
Sets the maximum number of periodic components in the vertical direction that are considered for the measurement. If the actual number of periodic components is higher than the set value, they are considered for the random jitter component.

Remote command:
`ADVJitter<m>:DCOMposition:NVERtical` on page 2509

(Other) bounded uncorrelated
If enabled, allows the selection of the (other) bounded uncorrelated jitter/noise: (O)BUJ, (O)BUJ(δ -δ) , RJ, (O)BUN, (O)BUN(δ -δ) , RN.

The **other** bounded uncorrelated jitter/noise includes the periodic jitter/noise components. It is displayed, if Periodic is enabled. Otherwise the bounded uncorrelated jitter/noise is displayed.

Remote command:
`ADVJitter<m>:DCOMposition:ENOBu` on page 2509

18.2.7 Saving the results

Once the advanced jitter analysis is enabled, you can save the results as a report or export the waveforms for future references.

Report

You can save a report of the advanced jitter analysis, that displays the "Adv. Jitter Results" table and the "Periodic components" table.

Advanced Jitter: Results of all Components

	StdDev	abs/pp	Min(StdDev)	Max(StdDev)	Min(abs/pp)	Max(abs/pp)	Wave Count
AdvancedJitterResults1	Analyzed source: Ch1Wfm1		Reference source: CDR	Use gate: Off			
Event Count	---	14.2 k	---	---	14.2 k	14.2 k	1
Random Jitter and Other Bounded Uncorrelated Jitter	54.9 ps	237 ps	54.9 ps	54.9 ps	237 ps	237 ps	1
Total Jitter (measured)	55.5 ps	196 ps	55.5 ps	55.5 ps	196 ps	196 ps	1
Data-Dependent Jitter	5.53 ps	53.5 ps	5.53 ps	5.53 ps	53.5 ps	53.5 ps	1
Periodic Jitter	8.4 ps	88.1 ps	8.4 ps	8.4 ps	88.1 ps	88.1 ps	1
Random Noise and Other Bounded Uncorrelated Noise	78.9 mV	605 mV	78.9 mV	78.9 mV	605 mV	605 mV	1
Total Noise	72.9 mV	486 mV	72.9 mV	72.9 mV	486 mV	486 mV	1
Data-Dependent Noise	61.8 mV	372 mV	61.8 mV	61.8 mV	372 mV	372 mV	1
Periodic Noise	18.8 mV	301 mV	18.8 mV	18.8 mV	301 mV	301 mV	1

Advanced Jitter: Periodic Components of Periodic Jitter

	Frequency	Value	Direction
AdvancedJitterPeriodicComponents1	Analyzed source: Ch1Wfm1		
1	206 MHz	13.1 ps	Horizontal (intrinsic)
2	210 MHz	13 ps	Horizontal (extrinsic)
3	207 MHz	9.94 ps	Horizontal (extrinsic)
4	190 MHz	9.57 ps	Horizontal (intrinsic)
5	202 MHz	9.23 ps	Horizontal (extrinsic)
6	3.1 GHz	9.32 mV	Vertical
7	3.19 GHz	8.64 mV	Vertical
8	2.16 GHz	8.54 mV	Vertical
9	2.08 GHz	8.36 mV	Vertical
10	2.92 GHz	7.87 mV	Vertical

See also Chapter 12.4, "Reports", on page 480.

Exporting the output signal

The following output signals from the advanced jitter analysis can be exported in the file formats `*.csv`, `*.bin` and `*.xml`:

- Step response
- Bathtub
- Histograms
- Tracks
- Spectrum

18.3 Clock data recovery

The process of clock data recovery (CDR) generates a reference clock from a high-speed serial data stream that is sent without a dedicated clock signal. The generated clock signal matches the frequency and is aligned to the phase of the data stream. The recovered clock can be used to sample the data stream, to obtain the sequence of transmitted bits, and to measure jitter parameters.

There are two ways to recover the clock signal:

- Using software algorithms that calculate the clock from data signal edges (option R&S RTO6-K12):
  Chapter 18.3.1, "Software CDR", on page 1112
- Using hardware CDR
  Option R&S RTO6-K13:Chapter 18.3.2, "Hardware CDR (option R&S RTO6-K13)", on page 1115

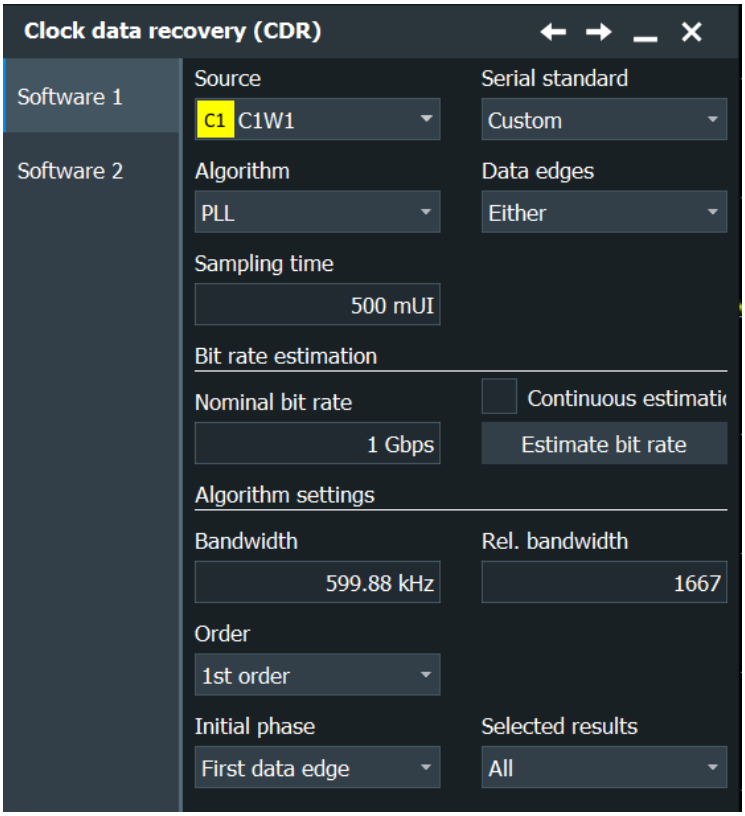
18.3.1 Software CDR

You can define two independent software CDR instances to recover clock signals. These clock signals are used for data timing measurements: time interval error, unit interval, and data rate (frequency of the clock signal). The lower the signal bandwidth and the higher the relative bandwidth, the longer is the settling time.

Software CDR is based on acquired data, needs a synchronization time to set up the clock. To get correct measurement results, the data acquired during synchronization time has to be discarded.

You can also display the recovered clock signal as a math waveform, see Chapter 18.3.3, "Displaying the recovered clock signal", on page 1120.

Access: [App Cockpit] > "Analysis" tab > "CDR" > "Software1"/"Software2"



Source
Selects the source of the data stream.

Remote command:
`CDR:SOFTware<m>:ESBRate:SOURce` on page 2511

Serial standard
For some serial standards, the bit rate and bandwidth are known to the instrument. Select the standard, and the "Nominal bit rate" and bandwidth settings are set automatically, no estimation is required.

Remote command:
`CDR:SOFTware<m>:SERStandard` on page 2510

Data edges
Selects the edges of the data stream that are used for the clock recovery.

- "Either": Both positive and negative edges are used
- "Positive / Negative": Only one edge direction is used. Use one of these settings if the other edge might deliver unreliable results.

Remote command:
`CDR:SOFTware<m>:ESLope` on page 2512

Algorithm
Sets the software algorithm that is used for clock data recovery.

"Constant freq." CDR uses the nominal bit rate to generate the clock signal. The method assumes that the frequency of the signal is constant during the complete acquisition.

"PLL" PLL is the phase-locked loop control system. It can follow slow deviations in the frequency of the data stream. Thus, it acts as highpass filter regarding the jitter that remains on the signal.

"Feed Forward" The feed forward algorithm is a mathematical optimization method used for clock data recovery. Feed forward is available if standards are of the listed serial used ("Custom"). It is faster than PLL, i.e., the settling time is shorter. Feed forward is also more precise than PLL. To set up the algorithm, set the bandwidth and the relative bandwidth.

Remote command:
`CDR:SOFTware<m>:ALGorithm` on page 2510

Sampling time
Sets a sampling time for the explicit clock signal, an offset for the clock edge in relation to the bit start. The clock edge sets the beginning of the unit interval. The sampling time is a number between 0 and 1. Value 0 sets the clock edge to the beginning of the bit period; value 0.5 sets the clock edge to the middle of the bit period.

Remote command:
`CDR:SOFTware<m>:SAMTime` on page 2515

Bit rate estimation
Bit rate estimation analyzes the source signal based on the given "Nominal bit rate" and corrects the value.

"Continuous" Enables ongoing bit rate correction.

"Estimate" Estimates the bit rate once, for the current acquisition.

Remote command:
`CDR:SOFTware<m>:ESBRate:CONTinuous` on page 2511
`CDR:SOFTware<m>:ESBRate:BREStimate` on page 2511

Nominal bit rate
Sets the quiescent frequency of the PLL. It corresponds to the data rate of the data stream from which the clock is to be recovered.

Remote command:
`CDR:SOFTware<m>:BITRate` on page 2512

Algorithm settings
Phase-locked loop parameters are listed below.

Note: Nomial bit rate, bandwidth and relative bandwidth are interacting settings. Modifying one parameter also changes one of the dependent parameters.

"Order" Sets the order of the PLL: first or second order. PLL of higher order can compensate for more complex jitter behavior.

"Damping" Sets the damping factor, which is only relevant for second order PLL.

"Bandwidth" Sets the PLL bandwidth. It defines the part of the spectrum that the PLL can follow during synchronization. The PLL bandwidth is usually defined by the transmission standard.

"Rel. bandwidth" Sets the relative bandwidth, that is the ratio of the nominal bit rate to the PLL bandwidth.

Remote command:
`CDR:SOFTware<m>:PLL:ORDer` on page 2513
`CDR:SOFTware<m>:PLL:BWIDth` on page 2514
`CDR:SOFTware<m>:PLL:RELBwidth` on page 2514
`CDR:SOFTware<m>:PLL:DAMPing` on page 2514

Reverse lock
Available for "Algorithm" > "PLL".

Enables the reverse lock PLL algorithm. The signal is locked on the first edge.

Remote command:
`CDR:SOFTware<m>:REVLockpll` on page 2514

Initial phase
Available for "Algorithm" > "PLL".

Defines the phase reference for the first clock edge.

"First sample" The first clock edge matches the first sample of the waveform at the left border of the display.

"First data edge" The first clock edge matches the first edge of the data signal.

Remote command:
`CDR:SOFTware<m>:SYNC` on page 2512

Selected results
Available for "Algorithm" > "PLL".

The PLL requires some time to synchronize to the phase of the data stream. You can select when the CDR algorithm returns clock edges:

"After initial sync." The clock edges of the synchronization time are discarded; results are gathered after initial synchronization of the CDR. Thus, meaningful TIE measurement results can be obtained.

"All" All clock edges are used.

Remote command:
`CDR:SOFTware<m>:RESults` on page 2513

18.3.2 Hardware CDR (option R&S RTO6-K13)

Hardware-based clock data recovery uses the digital CDR module to recover the clock edges. The generated clock is acquired similar to a waveform and occupies an adjacent analog channel. If the source is CH1, the recovered clock blocks CH2, and vice versa. CH3 and CH4 work in the same way. The occupied channel cannot be displayed, but the generated clock can be displayed as math waveform.

Hardware CDR is a continuous process, the recovered clock is always synchronized. HW CDR is the best way to analyze the jitter and eyes of continuous signals, and of buses that use Spread Spectrum Clocking. The complete data can be used, there is no synchronization time needed as with SW CDR, and no data must be discarded. Even short acquisitions are useable with HW CDR. For burst signals, HW CDR can be used for decoding, but it is inapplicable for eye and jitter measurement.

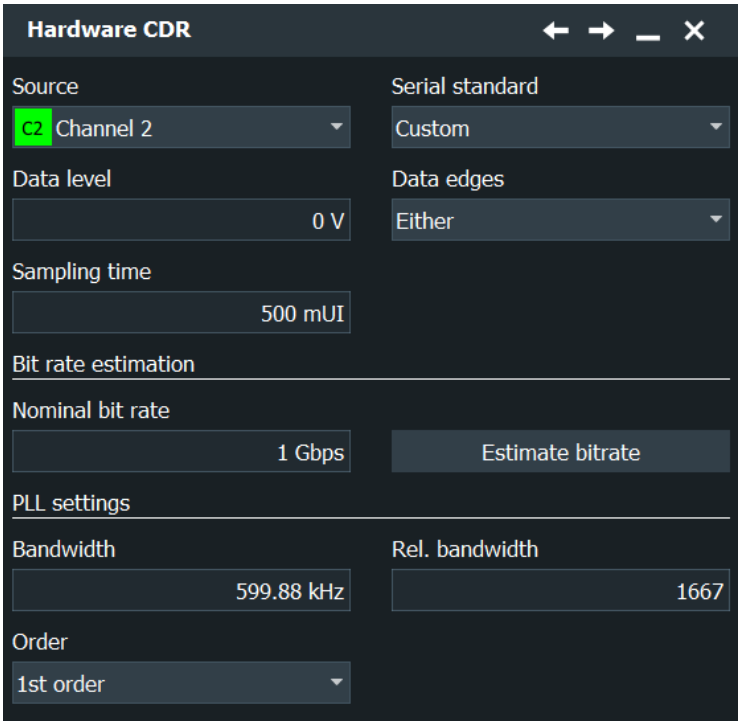
The recovered clock can be used:

- As trigger source for the CDR trigger, which is the basis for reliable jitter measurements, see Chapter 18.3.2.2, "CDR trigger", on page 1118.
- For the serial pattern trigger, see Chapter 18.3.2.3, "Serial pattern trigger using CDR", on page 1119.
- For TIE, unit interval and data rate measurements, see Chapter 18.1.2.3, "Data measurement settings", on page 1074.
- Display the generated clock as math waveform, see Chapter 18.3.3, "Displaying the recovered clock signal", on page 1120.

Jitter measurements and the recovered clock math waveform, which are based on HW CDR, interpret and display the acquired jitter data. If you change the HW CDR settings, a new acquisition is required. To acquire jitter data with HW CDR, the trigger type "CDR" or the trigger type "Serial pattern" with active CDR is required.

18.3.2.1 Hardware CDR setup

Access: "Menu" > "Trigger" > "Setup" tab > "Type = Serial Pattern" > "Hardware CDR"



Source
Selects the channel signal that is used for clock recovery.

The source cannot be changed if the CDR trigger is selected in the trigger setup. In this case, the instrument triggers on the recovered clock; trigger source and CDR source are the same.

Remote command:
`CDR:HARDware:SOURce` on page 2515

Serial standard
The following predefined standards are currently available:

Serial standard	Nom. bit rate
USB 2.0	480 Mbps
Displayport 1.1	1.296 Gbps
Fibre Channel 1x	1 Gbps
Fibre Channel 2x	2 Gbps
HDMI 1.2	1.32 Gbps
PCI-E_Gen_1	2.5 Gbps
SATA Gen 1 (1/500)	1.5 Gbps
SATA Gen 1 (1/1667)	3 Gbps

Remote command:
`CDR:HARDware:SERStandard` on page 2516

Data level
Sets the threshold to determine the position of rising and falling edges in the data input signal. The threshold level is also the trigger level for the specified input channel.

Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1391

Data edges
Selects the edges of the data stream that are used for the clock recovery.

- "Either": Both positive and negative edges are used
- "Positive / Negative": Only one edge direction is used. Use one of these settings if the other edge might deliver unreliable results.

Remote command:
`CDR:HARDware:ESLope` on page 2516

Nominal bit rate
Sets the quiescent frequency of the PLL. It corresponds to the data rate of the data stream from which the clock is to be recovered.

Remote command:
`CDR:HARDware:BITRate` on page 2516

Estimate
Estimates the bite rate once, for the current acquisition.

Remote command:
`CDR:HARDware:BREStimate` on page 2516

Sampling time
Sets a sampling time for the clock signal, an offset for the clock edge in relation to the bit start. The clock edge sets the beginning of the unit interval. The sampling time is a number between 0 and 1. Value 0 sets the clock edge to the beginning of the bit period; value 0.5 sets the clock edge to the middle of the bit period.

Remote command:
`CDR:HARDware:SAMTime` on page 2517

PLL settings
Phase-locked loop parameters are listed below.

Note: Nomial bit rate, bandwidth and relative bandwidth are interacting settings. Modifying one parameter also changes one of the dependent parameters.

"Bandwidth"	Sets the PLL bandwidth. It defines the part of the spectrum that the PLL can follow during synchronization. The PLL bandwidth is usually defined by the transmission standard.
"Rel. bandwidth"	Sets the relative bandwidth, that is the ratio of the nominal bit rate to the PLL bandwidth.
"Order"	Sets the order of the PLL: first or second order. PLL of higher order can compensate for more complex jitter behavior.
"Damping"	Sets the damping factor, which is only relevant for second order PLL.

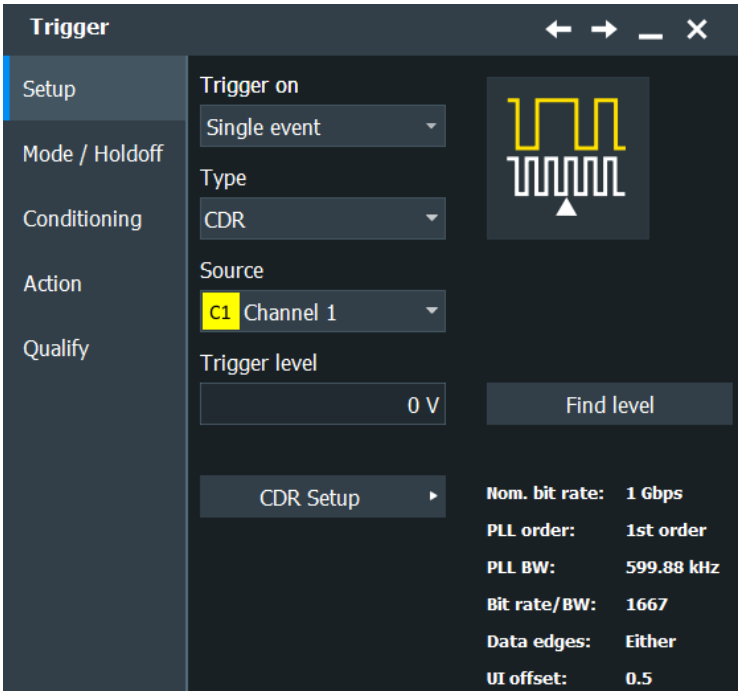
Remote command:
`CDR:HARDware:PLL:ORDer` on page 2516
`CDR:HARDware:PLL:BWIDth` on page 2517
`CDR:HARDware:PLL:RELBwidth` on page 2517
`CDR:HARDware:PLL:DAMPing` on page 2517

18.3.2.2 CDR trigger

The CDR triggers on the edges of the clock that is recovered from a data signal by the hardware CDR. The CDR trigger is the prerequisite for correct eye formation, and thus for correct jitter measurements and statistics.

The trigger point corresponds to the clock edge that indicates the bit start, and it considers the UI offset.

Access: directly from the hardware CDR setup, or "Menu" > "Trigger" > "Setup" tab > "Type" = *CDR*



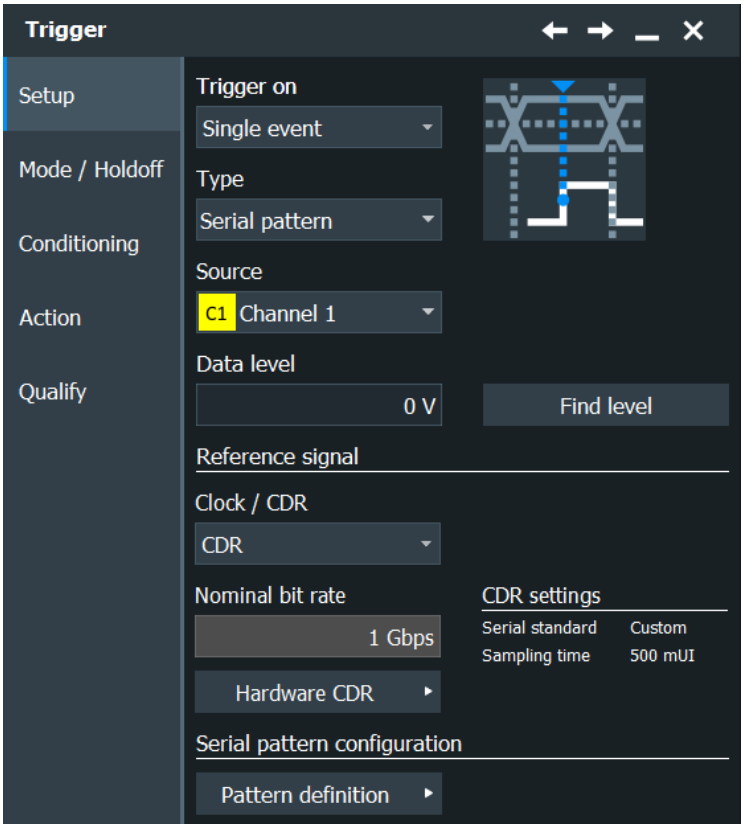
Trigger level
Sets the voltage level for the trigger event. For CDR, the trigger level is also the threshold level for the specified input channel.

Remote command:
`TRIGger<m>:LEVel<n>[:VALue]` on page 1391

18.3.2.3 Serial pattern trigger using CDR

If option R&S RTO6-K13 is installed, you can use the recovered clock signal for the serial pattern trigger instead of a real clock signal.

1. Select the trigger source and the trigger type "Serial pattern".
2. Set "Clock / CDR" to "CDR".
3. Tap "Hardware CDR" to set up the clock data recovery. See Chapter 18.3.2.1, "Hardware CDR setup", on page 1116 for details.



18.3.3 Displaying the recovered clock signal

The clock signal that is recovered by CDR can be displayed as a math waveform. The generated waveform is time-correlated to the data waveform.

1. Prerequisite: Configure the CDR method.
2. Press the [Math] key.
3. On the "Setup" tab, select the "Equation" subtab.
4. Double-tap the entry field to open the formula editor.
5. Tap "More", and again "More".
6. Tap "Clear".
7. Tap "CDR" and select the CDR method.
8. Complete the expression:
   SW CDR: e.g., CDR(sw1,Ch1Wfm1)
   HW CDR: CDR(hw)
9. Tap "Enter".
10. Enable the math signal.

19 Power analysis (option R&S RTO6-K31)

With the R&S RTO6 and option R&S RTO6-K31, you can perform power analysis measurements.

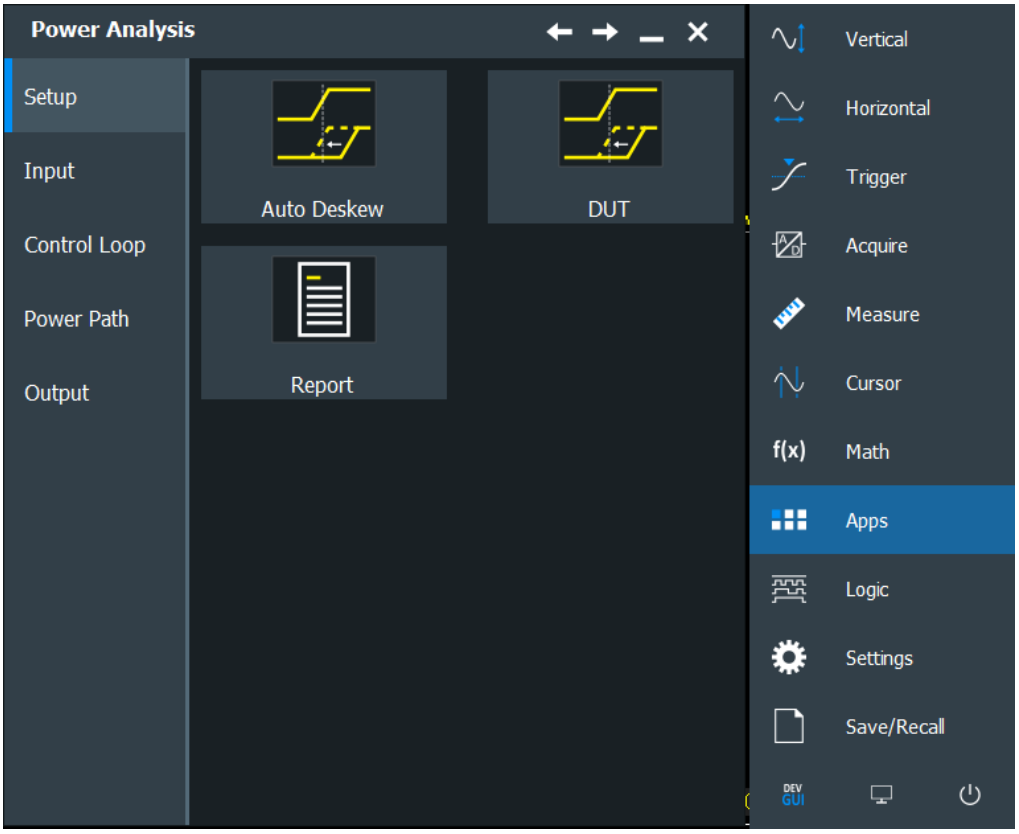
The following power measurements are available:

- Power quality
- Inrush current
- Current harmonic
- Modulation analysis
- Dynamic on resistance
- Slew rate
- Safe operating area (S.O.A.)
- Turn on/off
- Switching loss
- Power efficiency
- Output ripple
- Transient response
- Output spectrum

19.1 Power measurement selection

Access: [App Cockpit]> "Analysis" > "Power"

The "Power Menu" is the entry point to all power measurements and the general setting required for them.

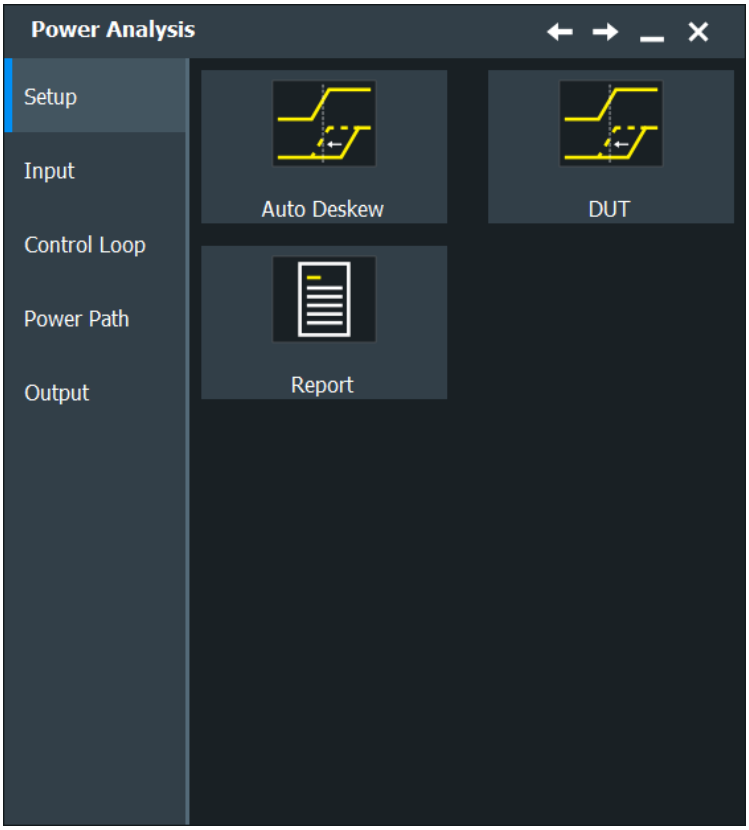


The dialog has several tabs:

- "Setup": general settings, that can be used by all measurements, like deskewing.
- "Input": measurements for performing input line analysis. They are used to measure the characteristics of the input power and the effects the power supply exudes to the input line.
- "Control Loop": measurements for characterizing the switching properties of a device.
- "Power Path": measurements for analyzing the behavior of the devices that control the power flow through the switched-mode power supply (SMPS) circuit, including switching devices and inductors.
- "Output": measurements for characterizing the behavior and quality of the SMPS output voltage.

19.1.1 Setup settings

Access: [App Cockpit]> "Analysis" > "Power" > "Setup"



The Setup settings, can be used by all measurements, like deskewing.

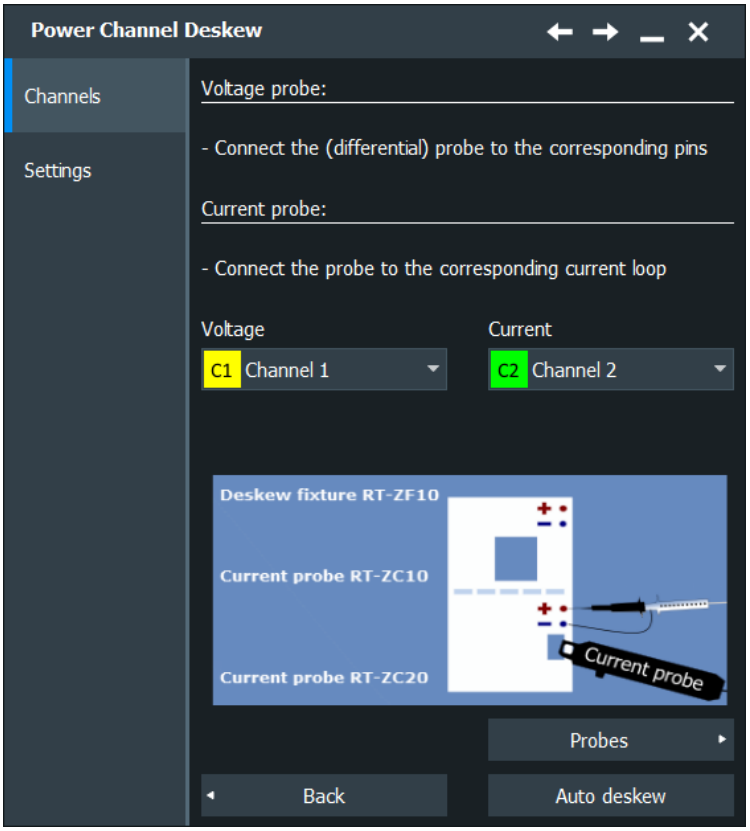
19.1.1.1 Auto deskew

The "Auto Deskew" dialog box guides you through the auto deskew of your current and voltage probes.

Required equipment:

- R&S RT-ZF20 power deskew fixture
- Rohde & Schwarz voltage probe
- Rohde & Schwarz current probe

1. Select "Analysis" > "Power".
2. Under "General", select "Auto Deskew".
3. Connect the voltage probe and the current probe to the oscilloscope.
4. Connect the probes to the R&S RT-ZF20 power deskew fixture as shown in the "Channels" tab:



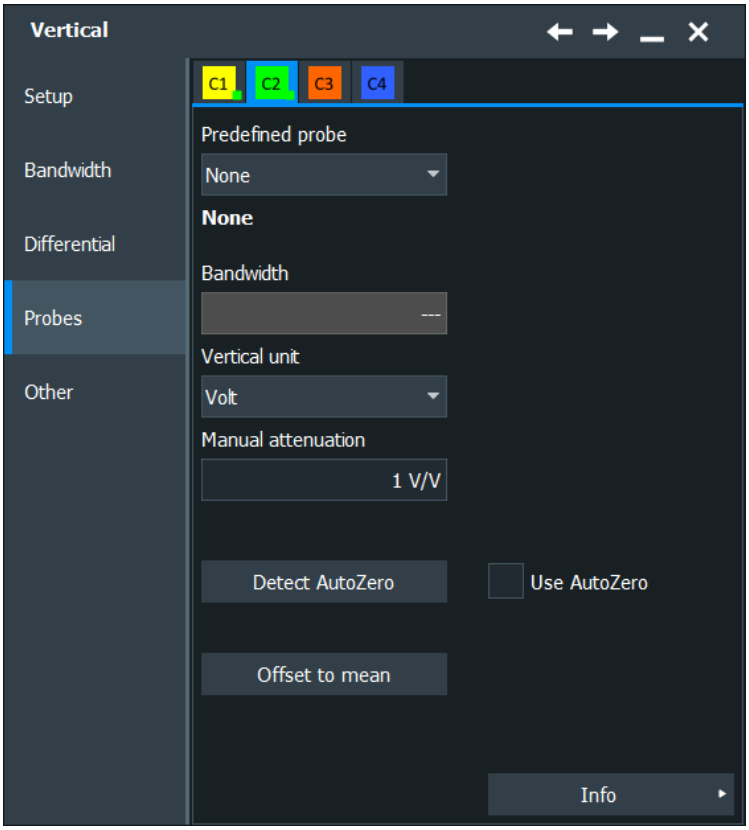
5. Select the correct channels for the "Current Source" and the "Voltage Source".
6. Tap "Next".
7. Check and complete the probe setup in the "Probes" tab.
   Current probes and high-voltage differential probes are not automatically detected by the instrument. Tap "Predefined probe" and select the correct probe type.
8. Tap the "Settings" tab.
9. Set the "Overwrite present skew setup" on page 1126 and "Activate user defined preset" on page 1127 options. These settings define whether the instrument uses the deskew result for user-defined preset and general skew settings.
10. Tap "Auto deskew".

    The probes are deskewed and the measurement can be started.

If no deskew fixture is available, you have to deskew your probes manually, see Chapter 5.3.1, "Skew", on page 123.

Probes

In the "Probes" tab, you check and set up your voltage and current probes.



Type, Name, Bandwidth
The fields show the characteristics of a recognized or predefined probe for information. If the instrument cannot recognize the probe, and the probe is not known, the "Type" is "None", and the other fields are empty.

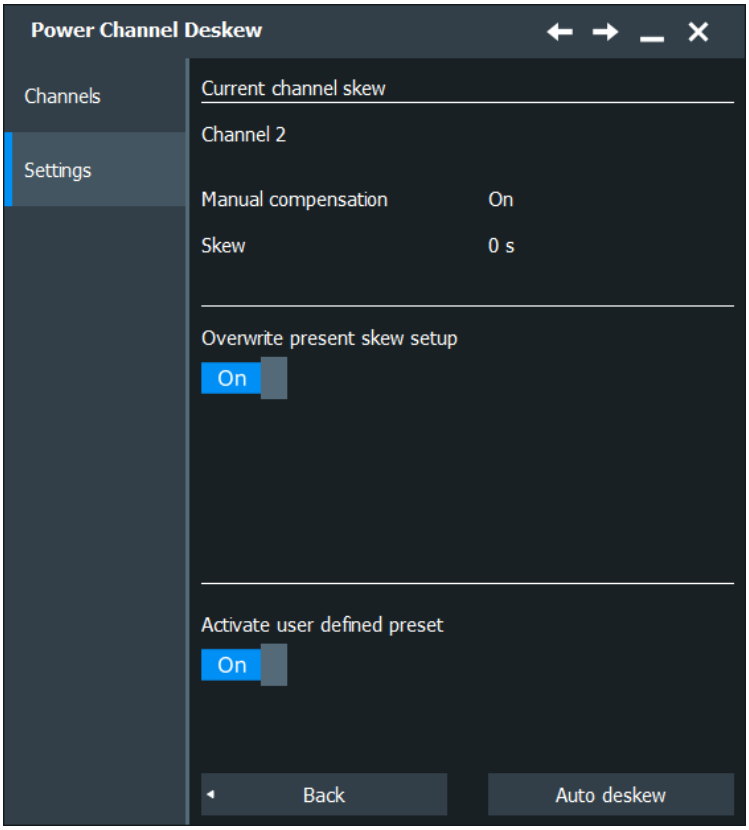
Remote command:
`PROBe<m>:SETup:TYPE?` on page 1344
`PROBe<m>:SETup:NAME?` on page 1344
`PROBe<m>:SETup:BANDwidth?` on page 1345

Predefined probe, Vertical unit, Attenuation
Current probes R&S RT-ZCxx, the high-voltage active probe R&S RT-ZD01 and the transmission line probe R&S RT-ZZ80 are not recognized automatically but the parameters of these probes are known to the instrument. Select the correct probe type and enter additional parameters if necessary. The corresponding "Vertical unit" and the "Attenuation" or "Gain" are set.

For an auto deskew, only Rohde & Schwarz probes are supported.

Settings

In this tab, you can define the preset behavior and how the auto deskew results are stored.



Current channel skew
Shows the skew settings of the channel connected to the current probe. Skew settings are defined in the "Horizontal > Skew" dialog box.

See also: Chapter 5.3.1, "Skew", on page 123

Overwrite present skew setup
If disabled, the instrument only stores the result of the auto deskew procedure as a separate value and does not use it. This value can be used later for power measurements. The general skew offset under "Current channel skew" remains unchanged.

If enabled, the result of the auto deskew procedure is used for all measurements on the selected channel. It is shown under "Current channel skew".

Remote command:
`POWer:DESKew:RESet` on page 2521

Auto deskew result
Available only if "Overwrite present skew setup" is disabled.

"Deskew value" Result of the auto deskew.

Remote command:
`POWer:DESKew:TIME?` on page 2521

"Apply" Writes the result of the auto deskew to the "Skew offset" of the selected channel.

Remote command:
`POWer:DESKew:CURRent` on page 2521

Activate user defined preset
If enabled, the deskew values are written to a user-defined preset file, and the user-defined preset is enabled. Thus, the probe setup and deskew values are not influenced by a manual [Preset].

See also: Chapter 4.6, "Preset setup", on page 98.

Remote command:
`POWer:DESKew:UDPReset` on page 2521

Auto Deskew
Starts an auto deskew.

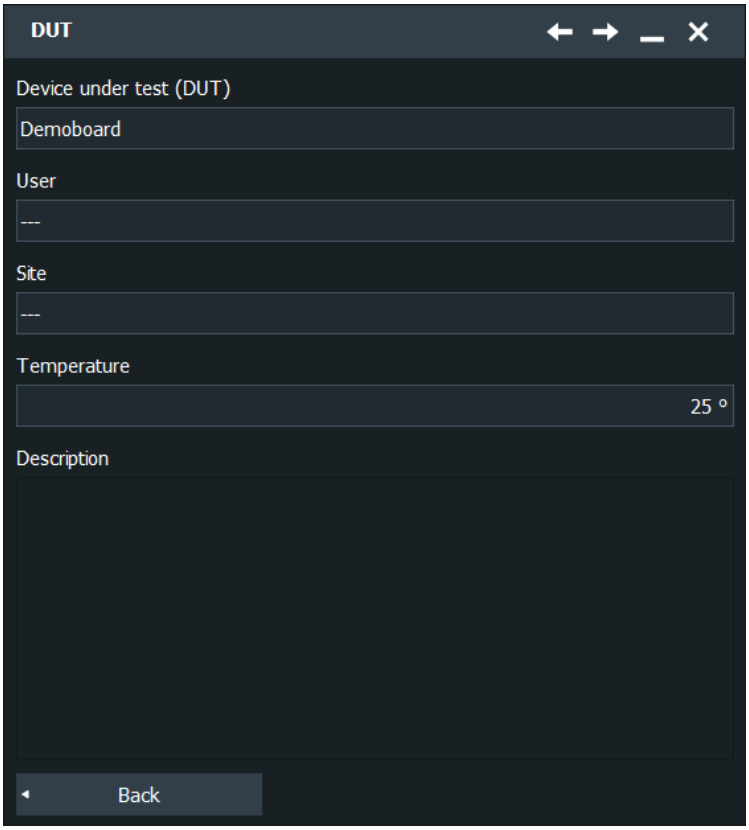
Make sure that the probes are configured correctly before you start the deskewing.

Remote command:
`POWer:DESKew:EXECute` on page 2521

19.1.1.2 DUT

Access: "Analysis" > "Power" > "DUT".

In this dialog, you can describe your device under test (DUT). The information set in this dialog can be used on the title page for a report generated from the "Power Analysis" measurements, see "Content" on page 1132.



Device under test (DUT)
Enter a name for your DUT.

Remote command:
`POWer:REPort:DUT` on page 2523

User
Enter a user.

Remote command:
`POWer:REPort:USER` on page 2523

Site
Enter a site.

Remote command:
`POWer:REPort:SITE` on page 2523

Temperature
Enter the temperature.

Remote command:
`POWer:REPort:TEMPerature` on page 2523

Description
Enter a description.

Remote command:
`POWer:REPort:DESCription` on page 2523

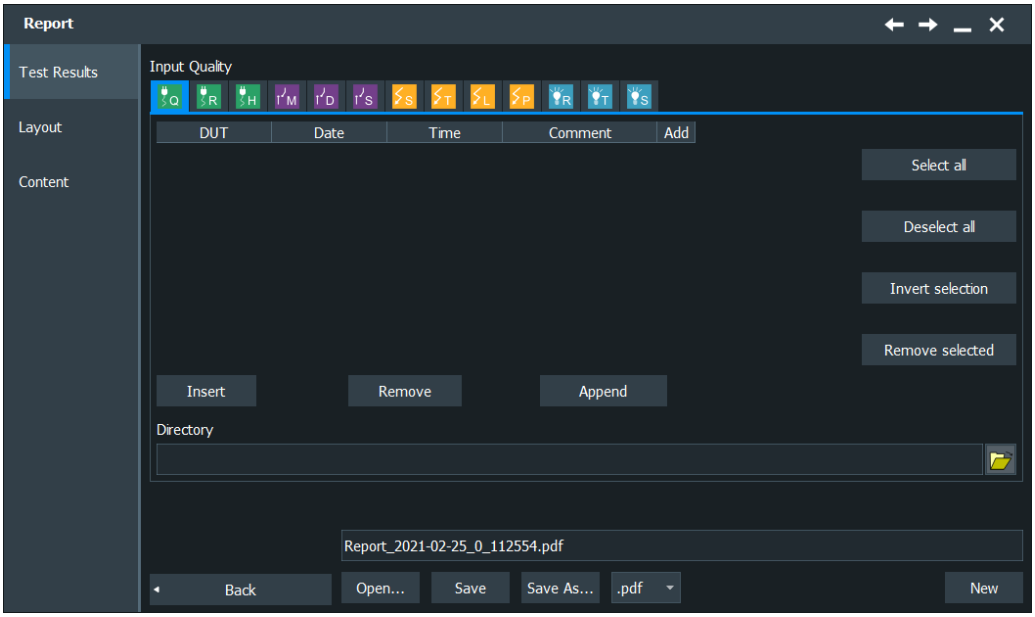
19.1.1.3 Report

Access "Analysis">"Power" > "Report".

Test results

After executing a measurement, you can press "Add to report" and save the results. In this tab, you can manage all saved measurement results.

At the top of the tab, you can switch through the different "Power Analysis" measurements.



Report Table
Shows a list of the available measurements.

After you select a Directory, you can manage previous report results from this directory. To add a measurement report press "Insert" or "Append". To remove a measurement report press "Remove".

"DUT"	Shows the name of the DUT, see Chapter 19.1.1.2, "DUT", on page 1127.
"Date"	Shows the date of the measurement.
"Time"	Shows at what time, the measurement result was added to report.
"Comment"	Enters a comment.
"Add"	Adds the selected measurement to the report.

Remote command:
`POWer:REPort:TEST:ADD` on page 2524
`POWer:REPort:TEST:COMMent` on page 2526

POWer:REPort:TEST:COUNt on page 2526
POWer:REPort:TEST:INSert on page 2524
POWer:REPort:TEST:LSENd? on page 2527
POWer:REPort:TEST:REMove on page 2524

Selection
Manages the selection of the result reports.

Select all ← Selection
Selects all result reports.

Remote command:
POWer:REPort:TEST:SEA on page 2525

Deselect all ← Selection
Deselects all result reports.

Remote command:
POWer:REPort:TEST:DSEA on page 2525

Invert Selection ← Selection
Inverts the selection of all result reports, meaning that all selected result reports are deselected and vice versa.

Remote command:
POWer:REPort:TEST:ISE on page 2525

Remove selected ← Selection
Removes the selected result report.

Remote command:
POWer:REPort:TEST:RSE on page 2525

Directory
Selects the directory, from which previous report results are inserted into the report table. You can use this directory to insert previously recorded report data into the current report.

Remote command:
POWer:REPort:TEST:DIRectory on page 2526

Report Path
Enter the file name to load or to save the report to, and select the file format with the format button on the right.

"Load"	Loads the most recently created report with the Windows default viewer application for the pdf/rtf file format.
"Open"	Opens a file selection dialog box and loads the selected file.
"Save"	Saves the data to the selected file.
"Save As..."	Opens the file selection dialog box and saves the data to the selected file.
".pdf/.rtf"	Selects the file format.

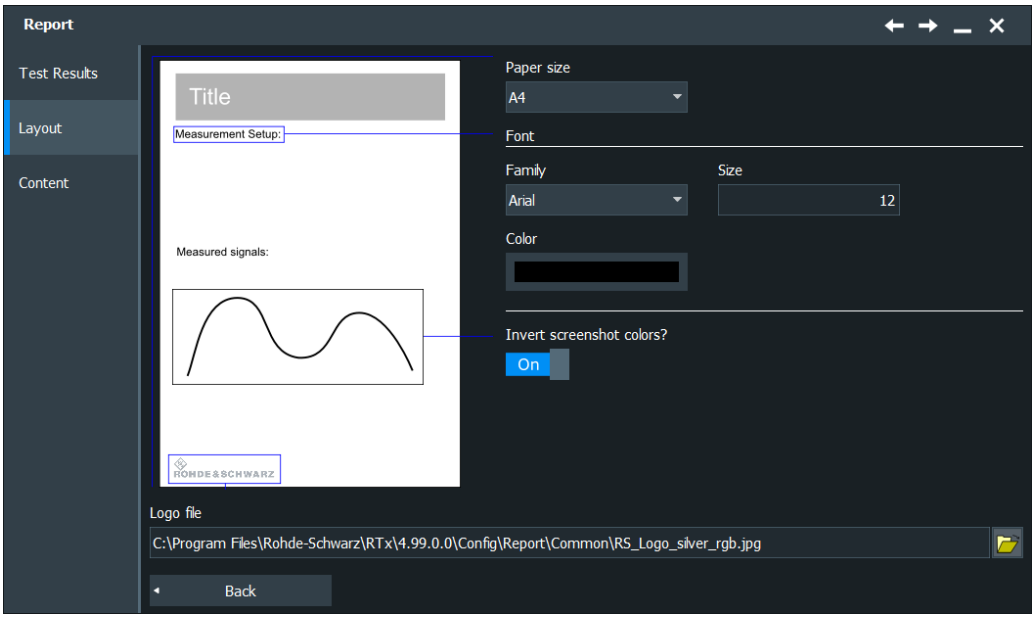
"New" Creates a new file.

"Delete" Deletes the selected file.

Remote command:
POWer:REPort:FILE:DELete on page 2524
POWer:REPort:FILE:NAME on page 2524
POWer:REPort:FILE:NEW on page 2524
POWer:REPort:FILE:SAVE on page 2524

Layout

In this tab, you can set up a layout for your report.



Paper size
Selects the paper size.

"A4" Selects A4.

"US Letter" Selects US Letter.

Remote command:
POWer:REPort:PAPersize on page 2524

Font
Sets the font for the report

Family ← Font
Selects the font family.

"Arial" Selects the font Arial.

"Helvetica" Selects the font Helvetica.

Remote command:
POWer:REPort:FONT:FAMI on page 2523

Size ← Font
Sets the font size.

Remote command:
`POWer:REPort:FONT:SIZE` on page 2523

Color ← Font
Sets the font color.

Remote command:
`POWer:REPort:FONT:COLO` on page 2523

Invert Screenshot Colors
Inverts the screenshot colors.

Remote command:
`POWer:REPort:INVert` on page 2525

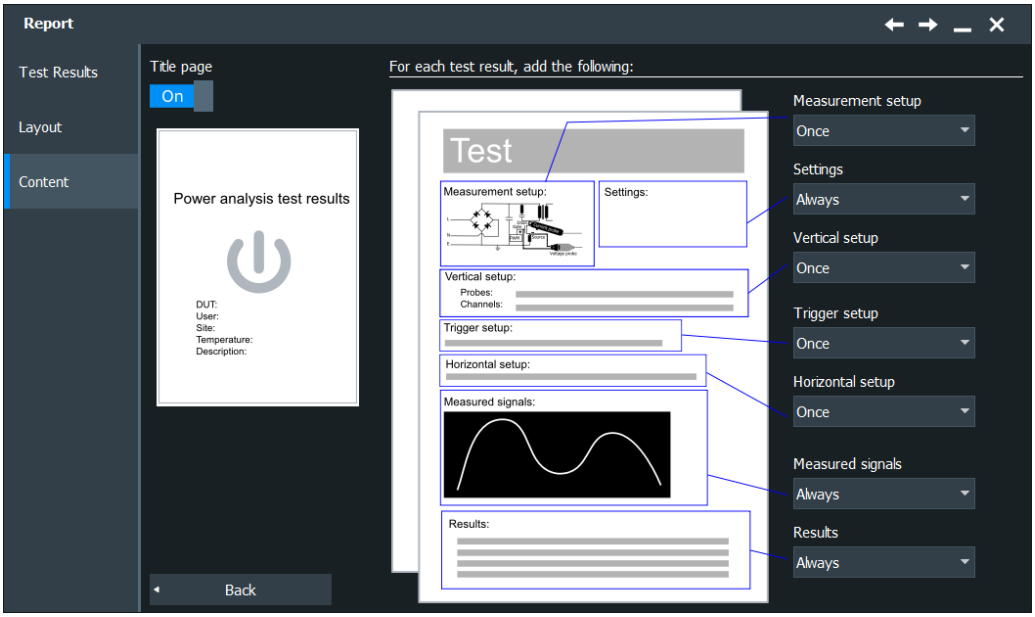
Logo File
Selects a path to a logo picture file.

Remote command:
`POWer:REPort:LOGO` on page 2523

Content

In this tab, you can select the contents of your report. For each content you can select how often it is included in the report:

- "Always": Shows the respective contents for each measurement.
- "Never": Does not show the respective contents in the report.
- "Once": Shows the respective contents once at the beginning of the report.



Title Page
Adds a Title page to the report. The contents can be set up in the "DUT" dialog, see Chapter 19.1.1.2, "DUT", on page 1127.

Remote command:
`POWer:REPort:CONTent:TITLe` on page 2523

Measurement Setup
Adds a graphic of the measurement setup.

Remote command:
`POWer:REPort:CONTent:MSETup` on page 2522

Settings
Adds the settings of the analysis.

Remote command:
`POWer:REPort:CONTent:SETTings` on page 2522

Vertical Setup
Adds the vertical setup settings.

Remote command:
`POWer:REPort:CONTent:VSETup` on page 2522

Trigger Setup
Adds the trigger setup settings.

Remote command:
`POWer:REPort:CONTent:TSETup` on page 2522

Horizontal Setup
Adds the horizontal setup settings.

Remote command:
`POWer:REPort:CONTent:HSETup` on page 2522

Measured signals
Adds a diagram of the measured signal.

Remote command:
`POWer:REPort:CONTent:MSIGnal` on page 2522

Results
Adds the result box.

Remote command:
`POWer:REPort:CONTent:RESU` on page 2522

19.2 Overview of power measurement setup

Each power analysis measurement dialog box consists of the following tabs:

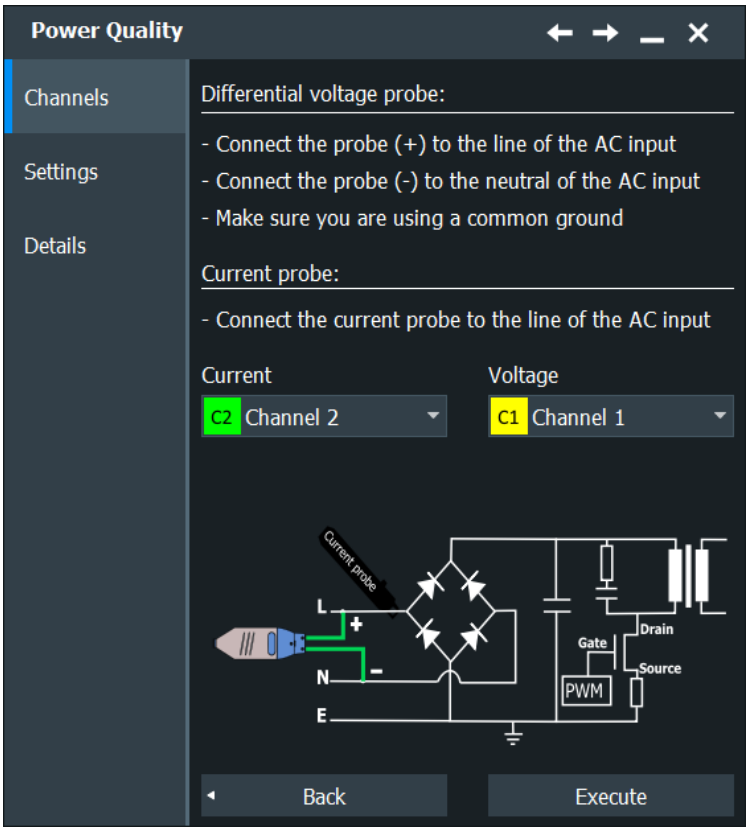
- Channels

- Settings
- Details

At the bottom of each tab, you can find two buttons. Tapping "Back", you can return to the power analysis measurement selection. "Execute" starts the power measurement.

19.2.1 Channels tab

In the "Channels" tab, you find information on the experimental setup of the selected power measurement. A short description explains what probes are needed and how to connect them. The description is supported by a block diagram of the experimental setup that shows the connection points for the probes.



Depending on the selected power measurement, one or two voltage sources and current sources are required.

Current Source
Sets the channel for the current source.

Remote command:
`POWer:SOURce:CURRent<1..2>` on page 2520

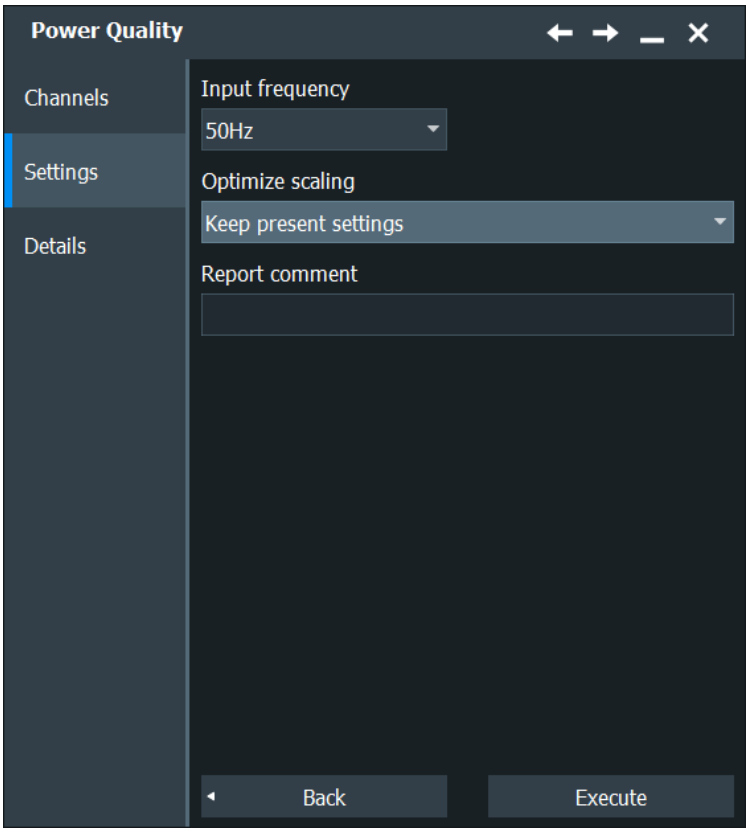
Voltage Source
Sets the channel for the voltage source input.

Remote command:
`POWer:SOURce:VOLTage<1..4>` on page 2520

19.2.2 Settings tab

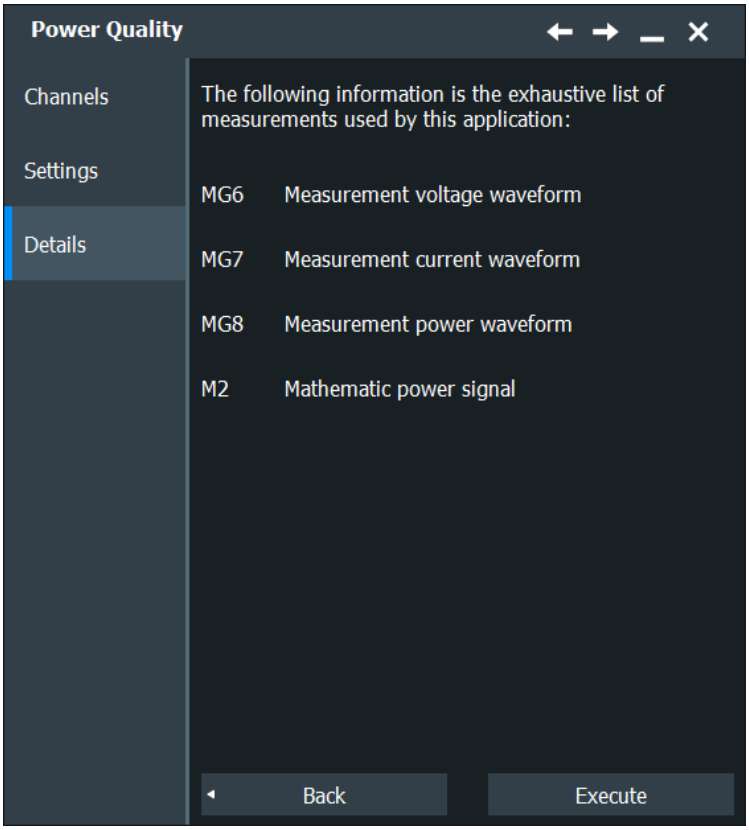
In the "Settings" tab, you configure the measurement and display settings. The settings depend on the selected power measurement.

For detailed information, see the "Settings" chapter of the relevant power measurement description.



19.2.3 Details tab

In the "Details" tab, you find information on the measurement resources used by the selected power measurement.



The following resources may be used:

- Measurements
- Mathematic waveforms
- Cursors
- XY-diagrams

The instrument enables the required resources when the power measurement is started.

19.3 Input

Access: [App Cockpit]> "Analysis" > "Power" > "Input"

The "Input" measurements are for performing input line analysis. They are used to measure the characteristics of the input power and the effects the power supply exudes to the input line.

19.3.1 Power quality

In an electric circuit power is a measure for the rate of flow of energy at a certain point of the circuit. The real power of a circuit, or the energy that can be used for work, is the portion of energy that is transferred in one direction over a complete cycle of the AC waveform. In AC circuits, however, inductive and capacitive elements can store energy temporarily. This portion of the power flow known as reactive power is then returned to the source without doing any work.

The " Power Quality" analysis measurements include the real power, the reactive power, the apparent power and the power factor. The crest factors and the phase angle between the current and voltage are also measured. These properties describe the power transfer in the system and allow you to characterize the power quality of the system.

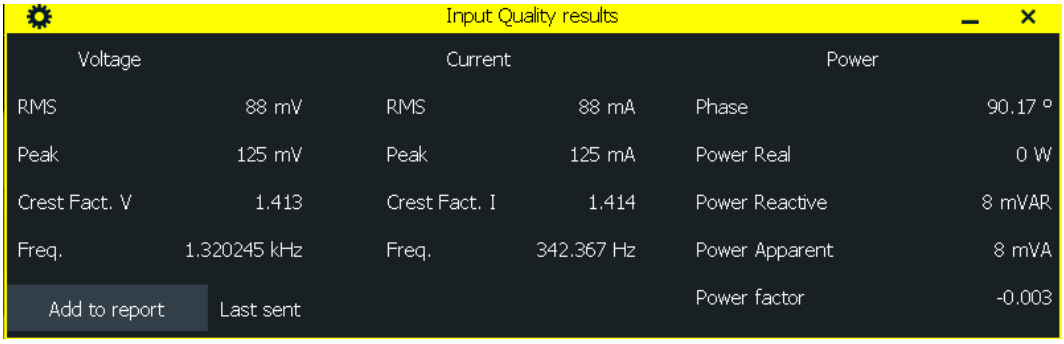
Required probes:

- Differential voltage probe
- Current probe

19.3.1.1 Power quality results

The results of " Power Quality" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  - the voltage waveform
  - the current waveform
  - the power waveform that is the product of the current and voltage waveforms
- The result box displays the numeric measurement results.



To measure and display the power quality, the instrument uses the following measurements and waveforms:

- "MG 6" to measure the voltage
- "MG 7" to measure the current
- "MG 8" to measure the power
- "M2" "Math 2" to calculate the power

The used resources are listed in the "Details" tab. See also: Chapter 19.2.3, "Details tab", on page 1135.

Voltage and current results

The voltage and current results are defined as follows:

Result	Description
RMS	Square root of the mean of the square of the current or voltage averaged over N cycles
Peak	Highest measured magnitude value of the voltage or current
Crest factor	Peak value / RMS value
Frequency	Frequency of the signal

Power results

The power in a system is described by several physical quantities: real power, reactive power, complex power, and phase angle. In Figure 19-1, you can see how these quantities are related if the voltage and the current are sinusoidal signals.

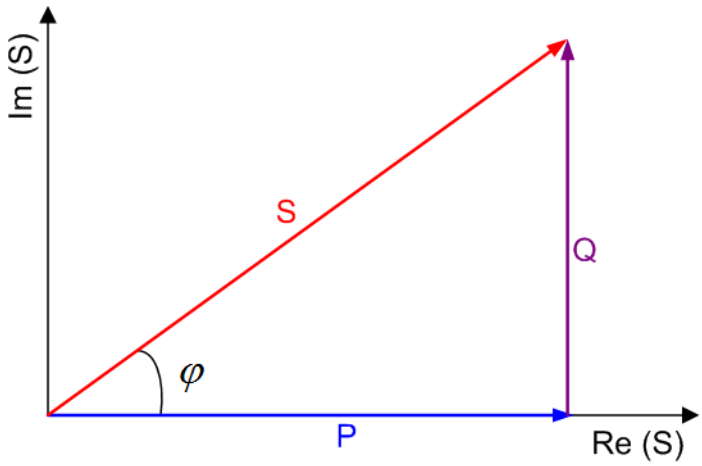


Figure 19-1: Power diagram for sinusoidal signals

P = real power [W]
Q = reactive power [VAR]
S = complex power [VA]
φ = phase angle between the current and the voltage sine waves [°]

The power results are defined as follows:

Result	Unit	Formula	Description
Power factor, P_{Factor}	-	$P_{Factor} = \|P\| / \|S\|$	Measure of the system efficiency. The value varies between -1 and 1.
Phase, φ	°	$\varphi = acos\ (P_{Factor})$	Phase angle between the current and the voltage sine waves.
Real power, P	W	$P = V_{INSTANTENEOUS} \bullet I_{INSTANTENEOUS}$ (averaged over N cycles)	Energy of the system that can be used to do work.
Reactive power, Q	VAR (Volt-Ampere reactive)	$Q = \|S\|\ sin\varphi$	Power flow that is temporally stored in a system because of the inductive and capacitive elements.
Apparent power, |S|	VA	$\|S\| = V_{RMS} \bullet I_{RMS}$ (averaged over N cycles)	|S| is the magnitude of the vector sum of real and reactive power (the complex power S).

The following remote commands are used for handling the measurement results:

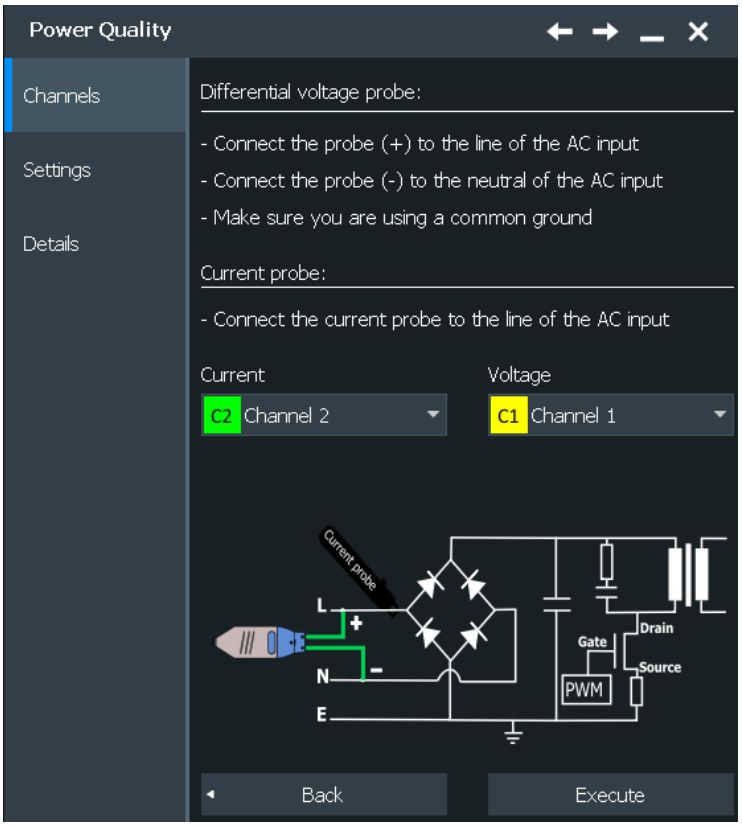
- `POWer:QUALity:RESult:CURRent:CRESTfactor?` on page 2528
- `POWer:QUALity:RESult:CURRent:FREQuency?` on page 2528
- `POWer:QUALity:RESult:CURRent:PEAK?` on page 2528
- `POWer:QUALity:RESult:CURRent:RMS?` on page 2528
- `POWer:QUALity:RESult:POWer:APParent?` on page 2528
- `POWer:QUALity:RESult:POWer:PFACtor?` on page 2528

- POWer:QUALity:RESult:POWer:PHASe? on page 2528
- POWer:QUALity:RESult:POWer:REACtive? on page 2528
- POWer:QUALity:RESult:POWer:REALpower? on page 2528
- POWer:QUALity:RESult:VOLTage:CREStfactor? on page 2528
- POWer:QUALity:RESult:VOLTage:FREQuency? on page 2528
- POWer:QUALity:RESult:VOLTage:PEAK? on page 2528
- POWer:QUALity:RESult:VOLTage:RMS? on page 2528
- POWer:QUALity:REPort:ADD on page 2528

19.3.1.2 Configuring power quality

For details of the configuration settings, see Chapter 19.3.1.3, "Power quality settings", on page 1141.

1. Select "Analysis" > "Power".
2. Under "Input", select "Power Quality".
3. Connect the differential voltage probe and the current probe to the oscilloscope.
4. Deskew the probes as described in Chapter 19.1.1.1, "Auto deskew", on page 1123.
5. Connect the probes to the DUT as shown in the "Channels" tab:



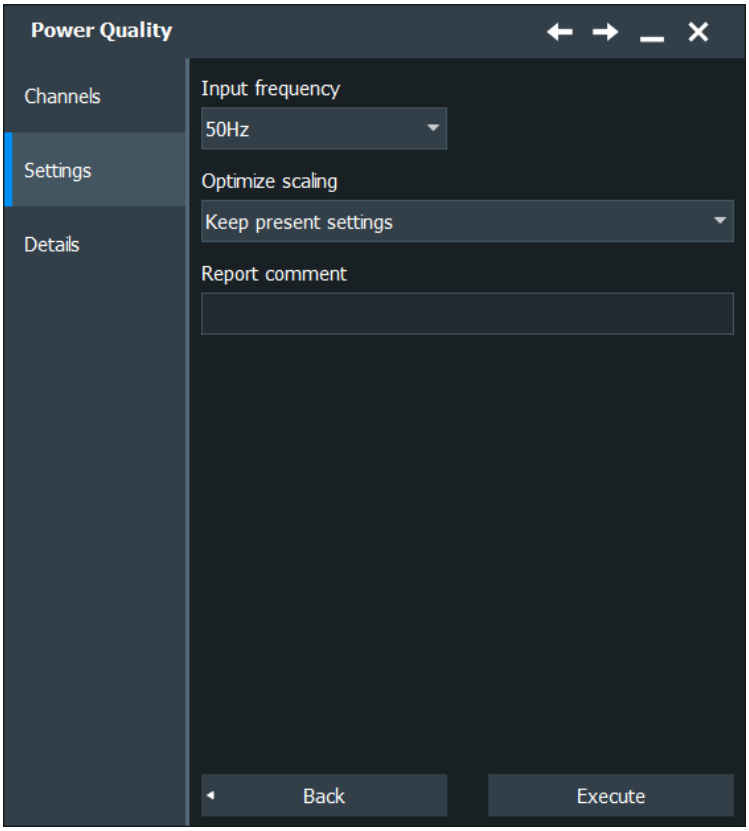
6. Select the correct channels of the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.
8. Set the "Input frequency" according to your signal.
9. Select an "Optimize Scaling" option.
10. Tap "Execute".

    On the screen you can see the measurement waveforms of the current, the voltage and the power. Also, the result box with numeric measurement results is shown. For details, see Chapter 19.3.1.1, "Power quality results", on page 1138.

19.3.1.3 Power quality settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 19.2.1, "Channels tab", on page 1134.

In the "Settings" tab, you configure the power measurement parameters and display settings.



Input frequency
Selects the input frequency of the source signal.

Remote command:
`POWer:QUALity:FREQ` on page 2527

Custom frequency
Sets the user-defined frequency if the "Input frequency" is set to "Custom".

Remote command:
`POWer:QUALity:FCUS` on page 2528

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:QUALity:AUTO` on page 2527

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the " Power Quality" measurement.

Remote command:
`POWer:QUALity:EXECute` on page 2527

19.3.2 Inrush current

The "Inrush Current" analysis measures the peak of the input current that is drawn by the device, when the device is first turned on.

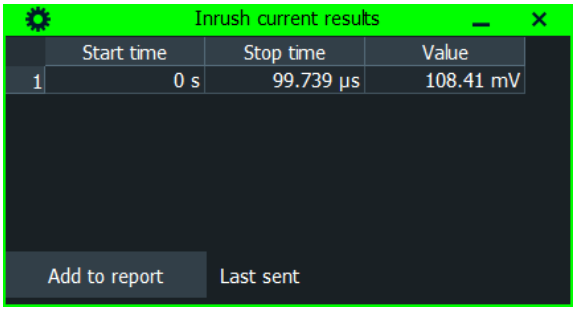
Required probes:

- Current probe

19.3.2.1 Inrush current results

The results of " Inrush Current" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  - the current waveform
- The result box displays the following numeric measurement results:
  - "Start time" / "Stop time" define the time period for the present gate
  - The "Value" stands for the maximum amplitude of the current for the present time period. This is the inrush current for the correspondent gate.



To measure and display the inrush current, the instrument uses the following measurements and waveforms:

- "MG 1" to "MG 5" to measure the inrush current of "Gate 1" to "Gate 5"

The used resources are listed in the "Details" tab. See also: Chapter 19.2.3, "Details tab", on page 1135.

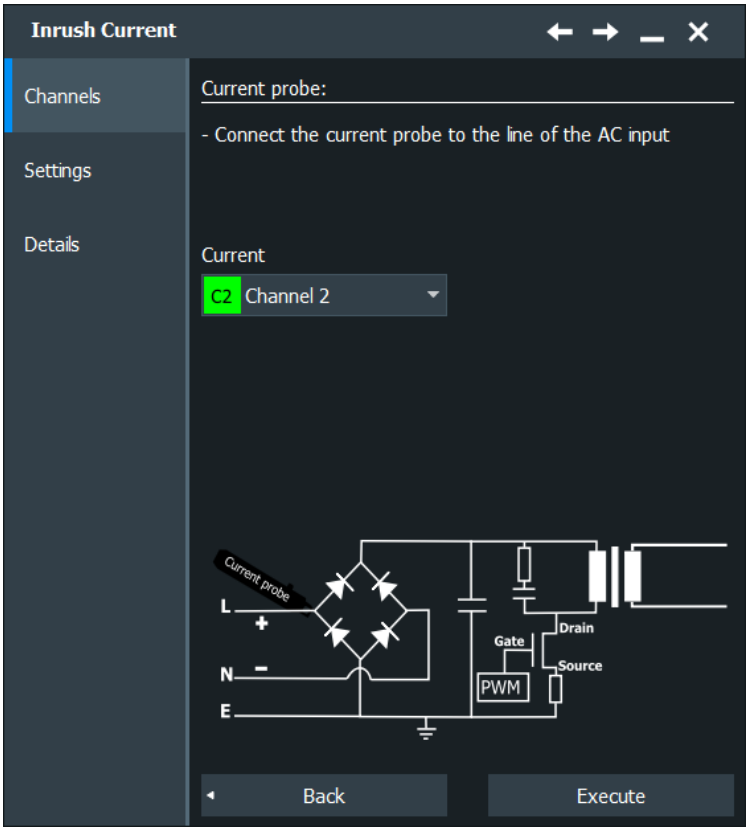
The following remote commands are used for handling the measurement results:

- `POWer:INRush:GATE<m>:VALue` on page 2530
- `POWer:INRush:REPort:ADD` on page 2531

19.3.2.2 Configuring inrush current

For details of the configuration settings, see Chapter 19.3.2, "Inrush current", on page 1142.

1. Select "Analysis" > "Power".
2. Under "Input", select "Inrush Current".
3. Connect the current probe to the oscilloscope.
4. Select the correct channel for the "Current Source".
5. Select "Vertical" > "Probe Setup" > "Channel" and set your probe parameters.
6. Connect the probes to the DUT as shown in the "Channels" tab:



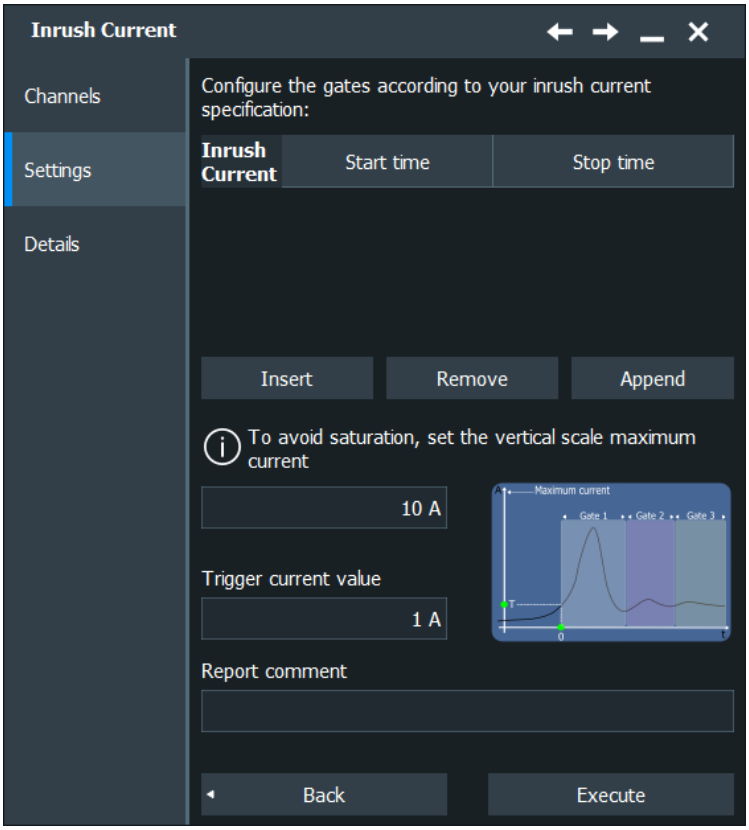
7. Select the "Settings" tab.
8. Set the "Trigger current value".
9. Set the "Maximum current" that should be displayed in the vertical scale.
10. In the "Gate configuration" table, define the different time periods. You can set up to five different gates.
11. Tap "Execute".
12. Start the DUT.

    On the screen, you can see the result box with the inrush current of each gate. For details, see Chapter 19.3.2.1, "Inrush current results", on page 1142.

19.3.2.3 Inrush current settings

In the "Channels" tab, you set the current source, see also: Chapter 19.2.1, "Channels tab", on page 1134.

In the "Settings" tab, you configure the inrush current measurement parameters and display settings.



Maximum current
Sets the maximum expected current for the vertical scale. Set the value according to your signal to avoid saturation.

Remote command:
`POWer:INRush:MAXCurrent` on page 2530

Trigger current value (T)
Sets the current value for the trigger. The measurement starts after the signal of the DUT reaches this current value.

Remote command:
`POWer:INRush:TRIGger` on page 2530

Gate Configuration
In this table, you can configure different gates (time periods). You can configure up to five different gates. The time periods of the defined gates may overlap.

To add a gate press "Insert" or"Append". To remove a gate press "Remove".

Remote command:
`POWer:INRush:ADD` on page 2529
`POWer:INRush:INSert` on page 2529
`POWer:INRush:REMove` on page 2529

Inrush current ← Gate Configuration
Shows the index of the gate.

Remote command:
`POWer:INRush:COUNt?` on page 2529

Start time ← Gate Configuration
Sets the start measuring time for the selected gate.

Remote command:
`POWer:INRush:GATE<m>:STARt` on page 2530

Stop time ← Gate Configuration
Sets the stop measuring time for the selected gate.

Remote command:
`POWer:INRush:GATE<m>:STOP` on page 2530

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the " Inrush Current " measurement.

Remote command:
`POWer:INRush:EXECute` on page 2530

19.3.3 Current harmonic

Current harmonics appear in an electric power system due to non-linear electric loads. The harmonics can be ejected back into the AC line and disturb other equipment on the grid. To avoid this disturbance, there are often standards of compliance that consumer or industry end-products should meet.

The "Current Harmonic" analysis tests the devices according to the pre-compliance standards EN 61000-3-2, MIL-STD-1399 and RTCA DO-160.

Required probes:

- Differential voltage probe
- Current probe

19.3.3.1 Current harmonic results

The results of "Current Harmonic" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  - the voltage waveform
  - the current waveform
  - the power waveform
- The result box displays a bar chart or a table with the numerical measurement results up to the 40th harmonics.
  To switch the display, tap "Show table" or "Show plot" accordingly

To measure and display the current harmonic, the instrument uses the following measurements and waveforms:

- "MG 6" to measure the power waveform
- "MG 7" to measure the spectrum voltage
- "MG 8" to measure the spectrum current
- "M2" Math 2 to calculate the power
- "M3" Math 3 to calculate the FFT of the voltage
- "M4" Math 4 to calculate the FFT of the current

The used resources are listed in the "Details" tab. See also: Chapter 19.2.3, "Details tab", on page 1135.

The current harmonic results are defined as follows:

Result Table	Bar Chart Match	Description
Harmonic Index	Value of the X-Axis	The harmonic order
Frequency	-	The frequency value of the signal
Value	Value of the Y-Axis. Shown by a green bar	The present value of the current harmonic
Maximum	Shown by a darkened green bar	The maximum measured value
Standard limit	Shown by a white bar	The maxim allowed value according to the selected standard
"Y max"	"Y max"	Sets the upper limit for the display of the Y scale. This value can be reset with the "Reset Max" button

The following remote commands are used for handling the measurement results:

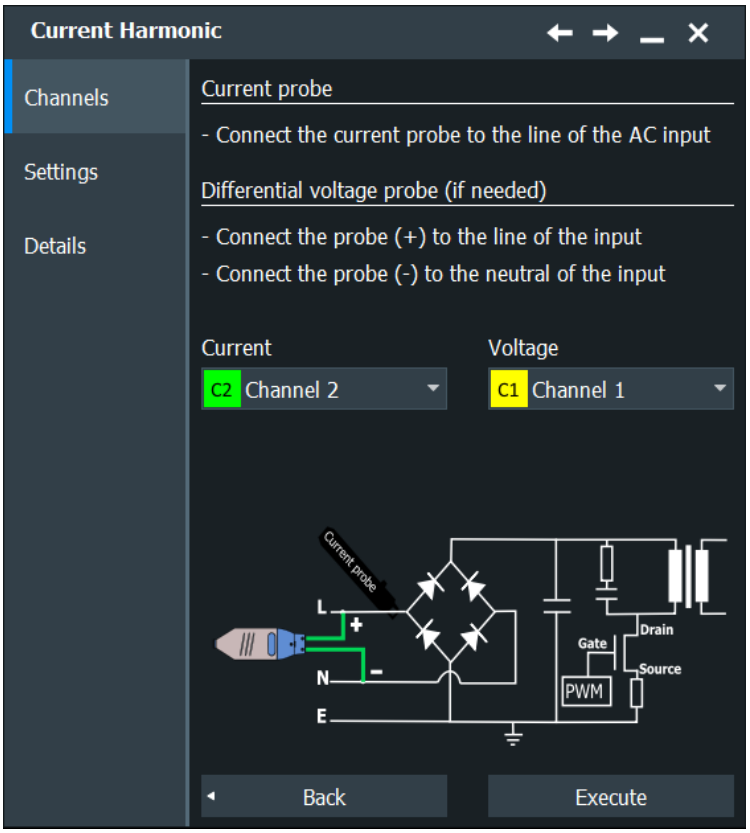
- `POWer:HARMonics:RESult<m>:FREQuency<n>:VALue?` on page 2532
- `POWer:HARMonics:RESult<m>:MAXValue<n>:VALue?` on page 2532
- `POWer:HARMonics:RESult<m>:STDinuse?` on page 2532
- `POWer:HARMonics:RESult<m>:STDValue<n>:VALue?` on page 2532
- `POWer:HARMonics:RESult<m>:VALue<n>:VALue?` on page 2532
- `POWer:HARMonics:REPort:ADD` on page 2532

19.3.3.2 Configuring current harmonic

For details of the configuration settings, see Chapter 19.3.3.3, "Current harmonic settings", on page 1148.

1. Select "Analysis">"Power".
2. Under "Power Analysis", select "Current Harmonic".
3. Connect the differential voltage probe and the current probe to the oscilloscope.

4. Deskew the probes as described in Chapter 19.1.1.1, "Auto deskew", on page 1123.
5. Connect the probes to the DUT as shown in the "Channels" tab:



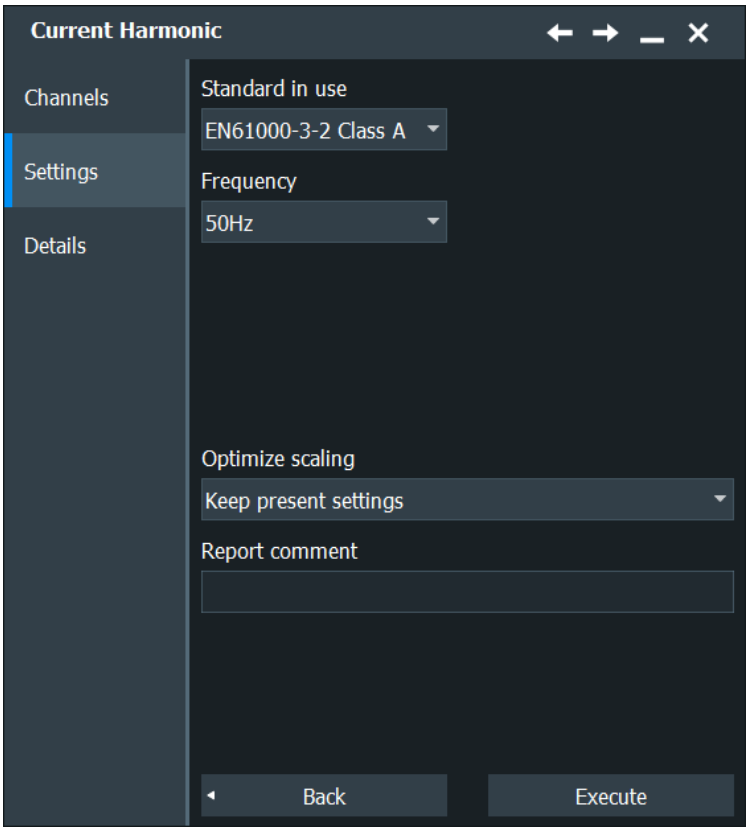
6. Select the correct channels for the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.
8. Set the "Standard in use"
9. Set the "Frequency" according to your signal.
10. Select an"Optimize Scaling" option.
11. Tap "Execute".

    On the screen you can see the measurement of the current, the voltage and the power. Also there is a table giving information about important measurement parameters. For details, see Chapter 19.3.3.1, "Current harmonic results", on page 1146.

19.3.3.3 Current harmonic settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 19.2.1, "Channels tab", on page 1134.

In the "Settings" tab, you configure the current harmonic measurement parameters and display settings.



Standard in use

Select the standard in use. For a list of the available standards, see Table 19-1.

Table 19-1: Current Harmonic pre-compliance standards

Standard	Application
EN 61000-3-2 Class A	Balanced 3-phase equipment, household appliances (excluding equipment identified as class D), tools (excluding portable tools), dimmers for incandescent lamps, audio equipment
EN 61000-3-2 Class B	Portable tools, not professional arc welding equipment
EN 61000-3-2 Class C	Lighting equipment
EN 61000-3-2 Class D	PC, PC monitors, radio, or TV receivers with an input power less than or equal to 600W
MIL-STD-1399	Military shipboard user equipment
RTCA DO-160	Environmental tests of avionics hardware

Remote command:
`POWer:HARMonics:STAN` on page 2533

Frequency
Selects the frequency of the input signal.

Remote command:
`POWer:HARMonics:ENFR` on page 2531
`POWer:HARMonics:MIFR` on page 2532
`POWer:HARMonics:DOFR` on page 2531

Revised Current
Available only for "Standard" > "RTCA DO-160".

Selects how the results are evaluated. Available are the following settings:

- Evaluation with current source only
- Evaluation with voltage source and revised current law
  - Display opposite voltage harmonic result chart
  - Do not display voltage result

Remote command:
`POWer:HARMonics:EVAL` on page 2532
`POWer:HARMonics:VOLT` on page 2533

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:HARMonics:AUTO` on page 2531

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Current Harmonics" measurement.

Remote command:
`POWer:HARMonics:EXECute` on page 2532

19.4 Control loop

Access: [App Cockpit]> "Analysis" > "Power" > "Control Loop"

The "Control Loop" measurements are used for characterizing the switching properties of a device.

19.4.1 Modulation analysis

The "Modulation Analysis " measures the control pulse signal to a switching device.

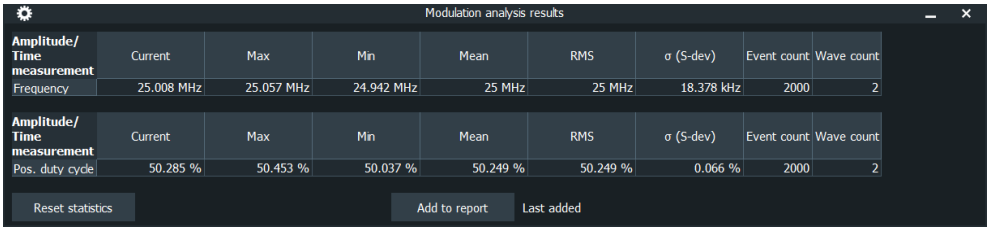
Required probes:

- Differential voltage probe
- Current probe

19.4.1.1 Modulation analysis results

The results of "Modulation Analysis " measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  - the voltage or the current waveform
  - for "Type >Turn on" a track of the frequency and the duty cycle.
- (Optional for "Type > Continuous" measurement) Two histograms display the density distribution of the measurement results in dependence of the frequency and the positive duty cycle.
- The result box displays the numeric measurement results.



To measure and display the power quality, the instrument uses the following measurements and waveforms:

- "MG 7" to measure the positive duty cycle
- "MG 8" to measure the frequency

The used resources are listed in the "Details" tab. See also: Chapter 19.2.3, "Details tab", on page 1135.

Table 19-2: Statistic result parameters

Label	Description
Current	Currently measured value
+Peak	Positive peak value (maximum)
-Peak	Negative peak value (minimum)
μ (Avg)	Average
RMS	Root mean square
σ (S-dev)	Standard deviation
Event count	Number of measured pulses
Wave count	Number of waveforms (acquisitions) the measurement is based on

"Modulation Analysis" is a statistical evaluation that is reset only if the measurement setup is changed or you reset the statistics.

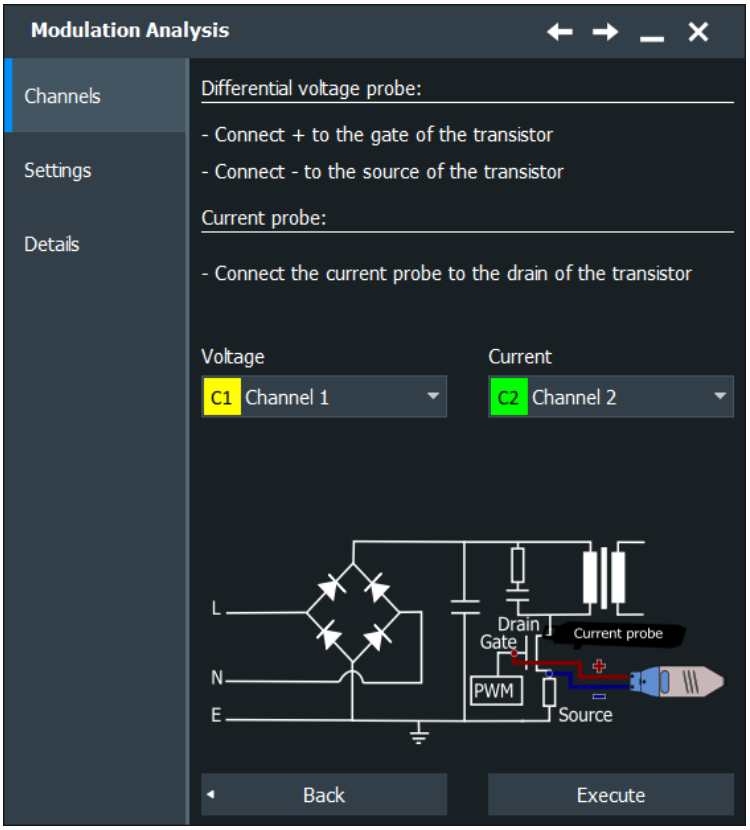
The following remote commands are used for handling the measurement results:

- `POWer:MODulation:RESult:ACTual?` on page 2534
- `POWer:MODulation:RESult:AVG?` on page 2534
- `POWer:MODulation:RESult:EVTCount?` on page 2534
- `POWer:MODulation:RESult:NPEak?` on page 2534
- `POWer:MODulation:RESult:PPEak?` on page 2534
- `POWer:MODulation:RESult:RMS?` on page 2534
- `POWer:MODulation:RESult:STDDev?` on page 2534
- `POWer:MODulation:RESult:WFMCount?` on page 2534
- `POWer:MODulation:REPort:ADD` on page 2534

19.4.1.2 Configuring modulation analysis

For details of the configuration settings, see Chapter 19.4.1.3, "Modulation analysis settings", on page 1154.

1. Select "Analysis" > "Power".
2. Under "Switching / Control Loop", select "Modulation Analysis".
3. Connect the differential voltage probe and the current probe to the oscilloscope.
4. Deskew the probes as described in Chapter 19.1.1.1, "Auto deskew", on page 1123.
5. Connect the probes to the DUT as shown in the "Channels" tab:



6. Select the correct channels for the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.
8. Set the "Source" and the "Type" of measurement.
9. Select an "Optimize Scaling" option.

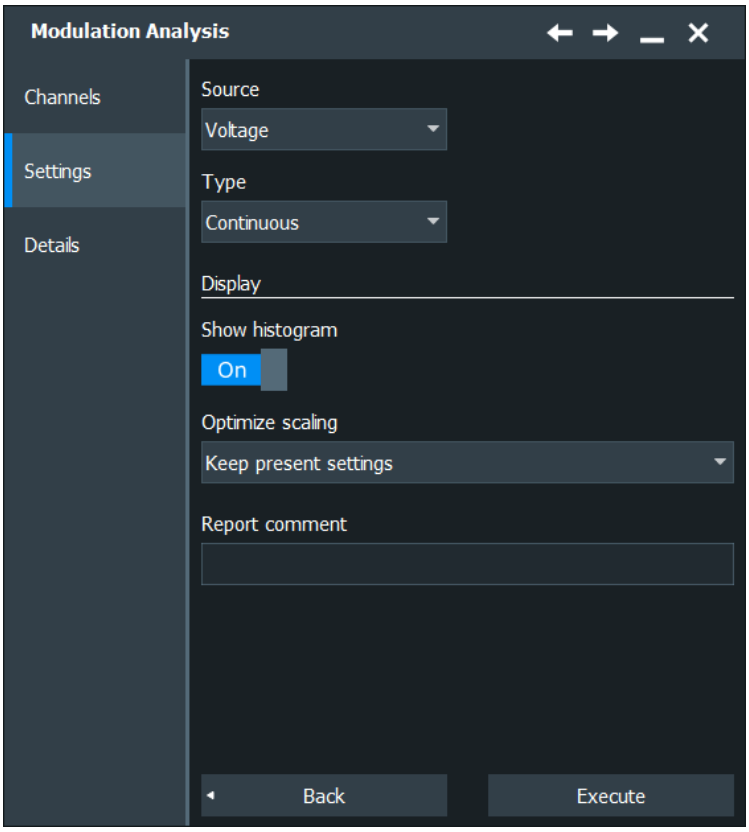
10. Tap "Execute".

    On the screen, you can see the measurement waveforms of the current or the voltage. Also, the result box with numeric measurement results is shown. For details, see Chapter 19.4.1.1, "Modulation analysis results", on page 1151.

19.4.1.3 Modulation analysis settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 19.2.1, "Channels tab", on page 1134.

In the "Settings" tab, you configure the modulation analysis parameters and display settings.



Source
Selects the source for the measurement.

Remote command:
`POWer:MODulation:SOURce` on page 2535

Type
Selects the type of signal flow for the measurement

"Continuous" The measurement is running continuously.

"Turn on" The measurement runs once when the DUT is turned on.

Remote command:
`POWer:MODulation:TYPE` on page 2535

Display Histogram
Available only for "Type" > "Continuous".

Enables the display of two histograms after the measurement is executed. The histograms show the density distribution of the measurement results in dependence of the frequency/ duty cycle in a graphic. Thus they illustrate the statistics of the measurements.

Remote command:
`POWer:MODulation:DHIStogram` on page 2534

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:MODulation:AUTO` on page 2534

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Modulation Analysis" measurement.

Remote command:
`POWer:MODulation:EXECute` on page 2534

19.4.2 Dynamic on resistance

The "Dynamic ON Resistance" analysis measures the resistance of a switching device, during operation. Because voltage and current may vary in time, the resistance is not constant, thus it is called dynamic ON resistance. It is defined as the ratio dV/dI.

The resistance-related voltage should be measured during a stable part of the switch node waveform, when the undershoot and ringing have decayed, after the high-to-low voltage transition.

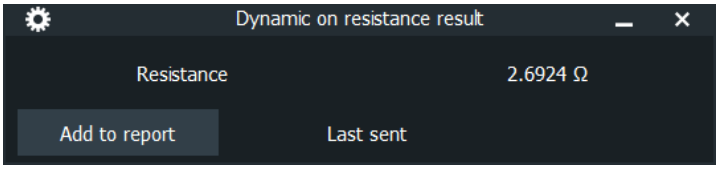
Required probes:

- Differential voltage probe
- Current probe

19.4.2.1 Dynamic on resistance results

The results of "Dynamic on Resistance" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  - the voltage waveform
  - the current waveform
- The result box displays the dynamic on resistance value.



To measure and display the dynamic on resistance, the instrument uses the following measurements and waveforms:

- "MG 5" to measure the amplitude of the voltage
- "MG 6" to measure the amplitude of the voltage
- "MG 7" to measure the amplitude of the current
- "MG 8" to measure the amplitude of the current
- "Cu 1 " Cursor 1 to measure gate ["t_0" ,"t_1"]
- "Cu 2 " Cursor 2 to measure gate ["t_2" ,"t_3"]

The used resources are listed in the "Details" tab. See also: Chapter 19.2.3, "Details tab", on page 1135.

The dynamic on resistance displayed as the result is defined as:

$$R = \frac{V(t_2) - V(t_0)}{I(t_3) - I(t_1)}$$

The points "t_0" , "t_1", "t_2" and "t_3" are defined by the cursor lines displayed in the result diagram of the measurement. You can move the cursor lines to define another area of interest.

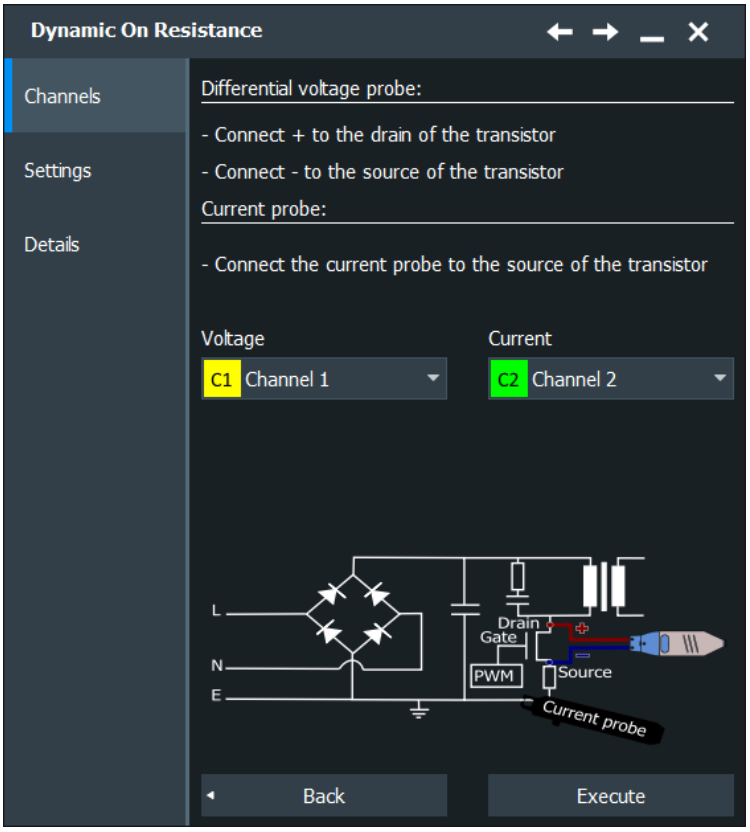
The following remote commands are used for handling the measurement results:

- `POWer:DONRes:RESult:RESistance?` on page 2536
- `POWer:DONRes:GATE<m>:STARt` on page 2536
- `POWer:DONRes:GATE<m>:STOP` on page 2536
- `POWer:DONRes:REPort:ADD` on page 2536

19.4.2.2 Configuring dynamic on resistance

For details of the configuration settings, see Chapter 19.4.2.3, "Dynamic on resistance settings", on page 1158.

1. Select "Analysis" > "Power".
2. Under "Switching / Control Loop", select "Dynamic on Resistance".
3. Connect the differential voltage probe and the current probe to the oscilloscope.
4. Deskew the probes as described in Chapter 19.1.1.1, "Auto deskew", on page 1123.
5. Connect the probes to the DUT as shown in the "Channels" tab:



6. Select the correct channels for the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.
8. If "Average" is selected, enter the "Average count", that is the number of waveforms used for average calculation.
9. Set the reset condition for the average calculation:
   - If "Time" is selected, enter the "Reset time".
   - If "Waveforms" is selected, enter the "Reset count".
10. Select an "Optimize Scaling" option.
11. Tap "Execute".

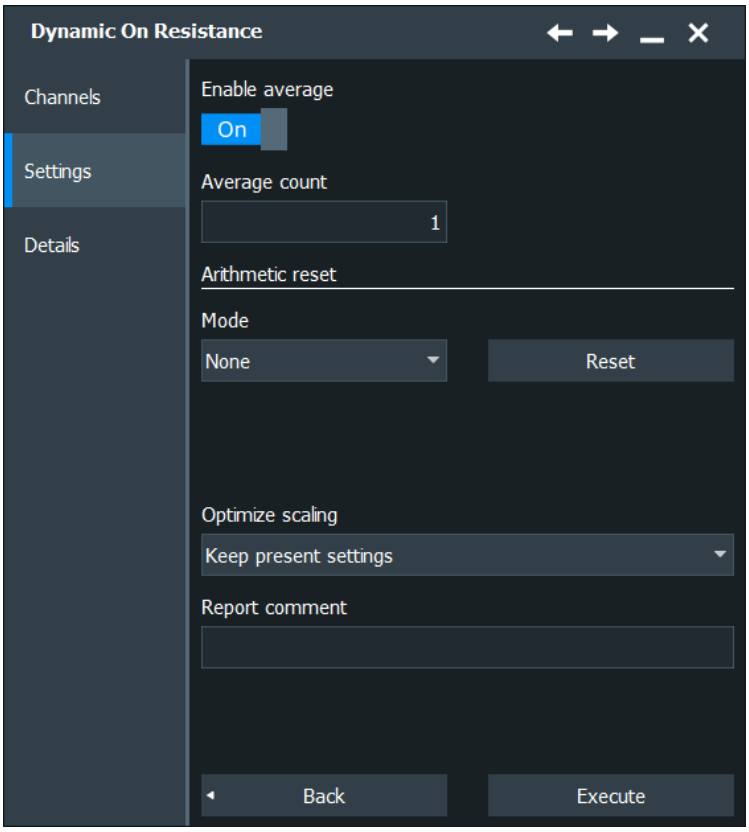
12. If needed adjust the cursors manually.

    On the screen, you can see the measurement waveforms of the current and the voltage. Also, the result box displays the dynamic on resistance. For details, see Chapter 19.4.2.1, "Dynamic on resistance results", on page 1156.

19.4.2.3 Dynamic on resistance settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 19.2.1, "Channels tab", on page 1134.

In the "Settings" tab, you configure the power measurement parameters and display settings.



Average
Enables the "Average" method for building the resulting waveform. The average is calculated from the data of the current acquisition and several acquisitions before. The method reduces random noise and other heterodyne signals. It requires a stable, triggered and periodic signal for correct function. The number of acquisitions for average calculation is defined with "Average count", and the "Reset mode" defines the restart condition.

Remote command:
`POWer:DONRes:AVG` on page 2535

Arithmetic Reset > Mode
Defines when the envelope and average evaluation restarts.

"None" No restart, the number of acquisitions considered by the waveform arithmetics is not limited.

"Time" Restarts the envelope and average calculation after the time defined in "Time".

"Waveforms" Restarts the envelope and average calculation after a number of acquired waveforms defined in "Count".

Remote command:
`ACQuire:ARESet:MODE` on page 1335
`ACQuire:ARESet:TIME` on page 1335
`ACQuire:ARESet:COUNt` on page 1335

Reset
Forces the immediate restart of the envelope and average calculation for all waveforms.

Remote command:
`ACQuire:ARESet:IMMediate` on page 1335

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:DONRes:AUTO` on page 2535

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Dynamic on Resistance" measurement.

Remote command:
`POWer:DONRes:EXECute` on page 2536

19.4.3 Slew rate

The "Slew Rate" analysis measures the rate of change of the voltage or current waveform during the switching of the switching transistor.

Required probes:

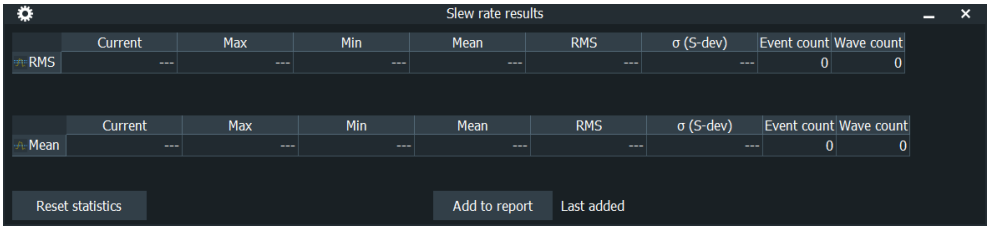
- Differential voltage probe

- Current probe

19.4.3.1 Slew rate results

The results of "Slew Rate" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  - the voltage waveform
  - the current waveform
  - a waveform of the derivative of voltage and current
- The result box displays the numeric measurement results. For a detailed description, see Table 19-2.



To measure and display the slew rate, the instrument uses the following measurements and waveforms:

- "MG 8" to measure the amplitude of the current or voltage waveform
- "M2" Math 4 to calculate the time derivative of the current or voltage waveform
- "Cu 1" Cursor 1 to determine the measurement area

The used resources are listed in the "Details" tab. See also: Chapter 19.2.3, "Details tab", on page 1135.

The following remote commands are used for handling the measurement results:

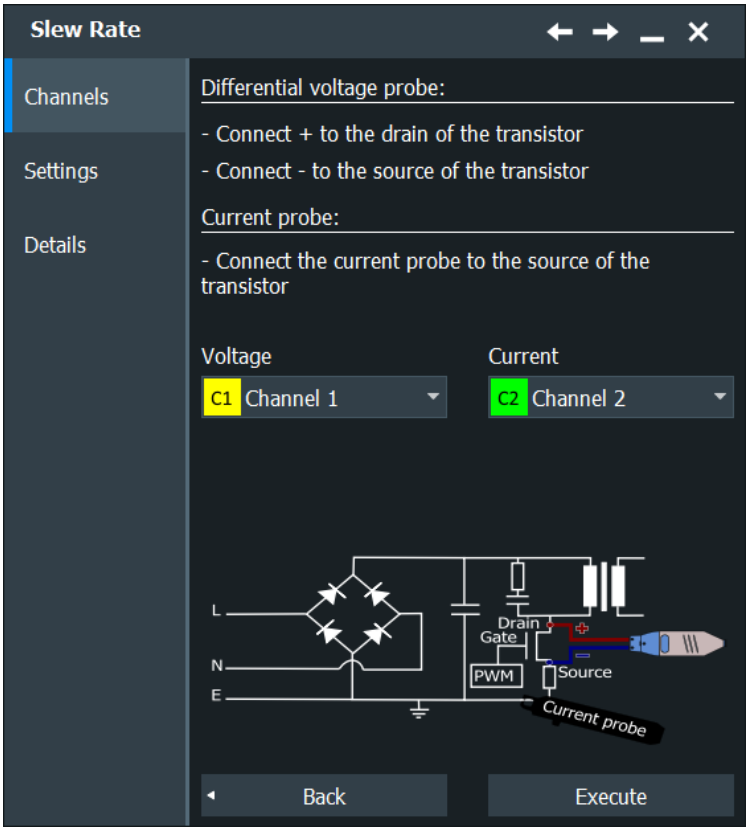
- `POWer:SLEWrate:RESult:ACTual?` on page 2538
- `POWer:SLEWrate:RESult:AVG?` on page 2538
- `POWer:SLEWrate:RESult:EVTCount?` on page 2538
- `POWer:SLEWrate:RESult:NPEak?` on page 2538
- `POWer:SLEWrate:RESult:PPEak?` on page 2538
- `POWer:SLEWrate:RESult:RMS?` on page 2538
- `POWer:SLEWrate:RESult:STDDev?` on page 2539
- `POWer:SLEWrate:RESult:WFMCount?` on page 2539
- `POWer:SLEWrate:REPort:ADD` on page 2538

19.4.3.2 Configuring slew rate

For details of the configuration settings, see Chapter 19.4.3.3, "Slew rate settings", on page 1162.

1. Select "Analysis" > "Power".

2. Under "Switching / Control Loop", select "Slew Rate".
3. Connect the differential voltage probe and the current probe to the oscilloscope.
4. Deskew the probes as described in Chapter 19.1.1.1, "Auto deskew", on page 1123.
5. Connect the probes to the DUT as shown in the "Channels" tab:



6. Select the correct channels for the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.
8. Select the "Source".
9. If "Average" is selected, enter the "Average count", that is the number of waveforms used for average calculation.
10. Set the reset condition for the average calculation:
    - If "Time" is selected, enter the "Reset time".
    - If "Waveforms" is selected, enter the "Reset count".
11. Select an "Optimize Scaling" option.

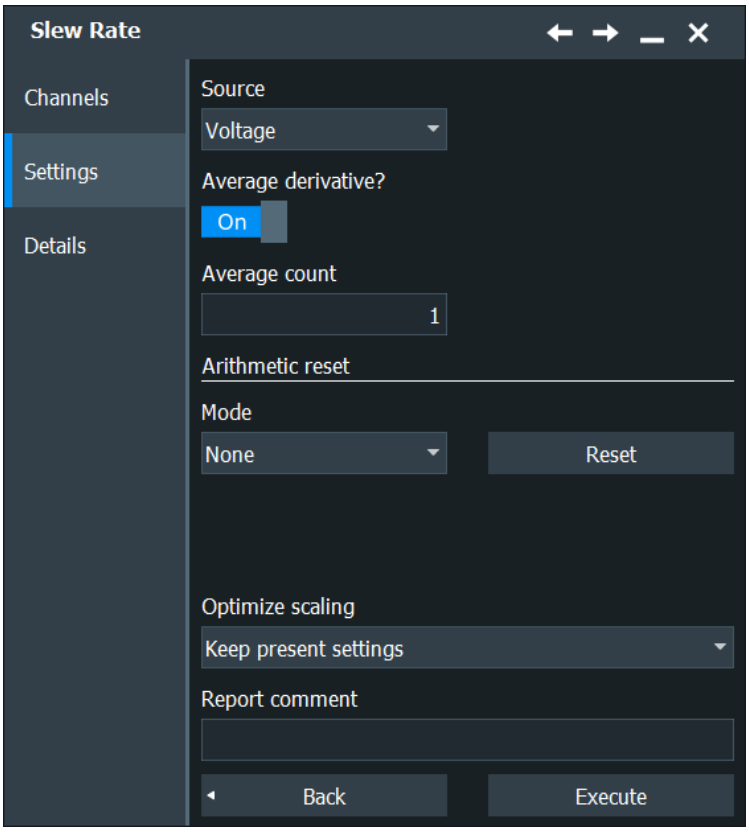
12. Tap "Execute".

    On the screen you can see the measurement waveforms of the slew rate, the current and the voltage. The result box with numeric measurement results is shown. For details, see Chapter 19.4.3.1, "Slew rate results", on page 1160.

19.4.3.3 Slew rate settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 19.2.1, "Channels tab", on page 1134.

In the "Settings" tab, you configure the slew rate measurement parameters and display settings.



Source
Selects dV/dt or dI/dt as the source of the measurement.

Remote command:
`POWer:SLEWrate:SOURce` on page 2537

Average
Enables the "Average" method for building the resulting waveform. The average is calculated from the data of the current acquisition and several acquisitions before. The method reduces random noise and other heterodyne signals. It requires a stable, triggered and periodic signal for correct function. The number of acquisitions for average calculation is defined with "Average count", and the "Reset mode" defines the restart condition.

Remote command:
`POWer:SLEWrate:AVGDeriv` on page 2537

Arithmetic Reset > Mode
Defines when the envelope and average evaluation restarts.

"None" No restart, the number of acquisitions considered by the waveform arithmetics is not limited.

"Time" Restarts the envelope and average calculation after the time defined in "Time".

"Waveforms" Restarts the envelope and average calculation after a number of acquired waveforms defined in "Count".

Remote command:
`ACQuire:ARESet:MODE` on page 1335
`ACQuire:ARESet:TIME` on page 1335
`ACQuire:ARESet:COUNt` on page 1335

Reset
Forces the immediate restart of the envelope and average calculation for all waveforms.

Remote command:
`ACQuire:ARESet:IMMediate` on page 1335

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:SLEWrate:AUTO` on page 2537

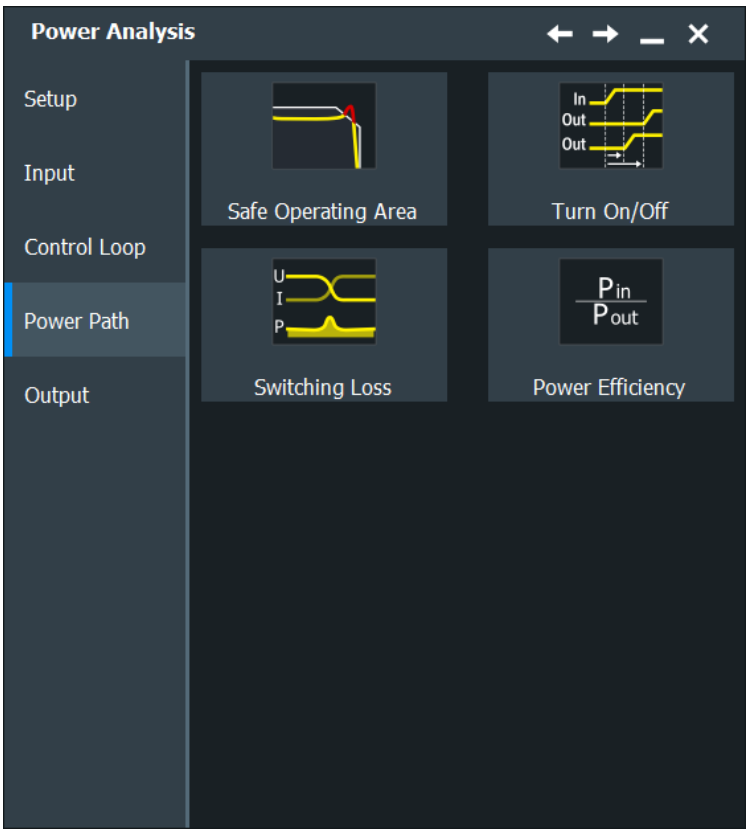
Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Slew Rate" measurement.

Remote command:
`POWer:SLEWrate:EXECute` on page 2537

19.5 Power path

Access: [App Cockpit]> "Analysis" > "Power" > "Power Path"



The "Power Path" measurements are used for analyzing the behavior of the devices that control the power flow through the switched-mode power supply (SMPS) circuit, including switching devices and inductors.

19.5.1 Safe operating area (S.O.A.)

The safe operating area is defined by the voltage and current conditions over which a power semiconductor device is expected to operate without self-damage. The "Safe Operating Area" analysis provides a diagram of the safe operating conditions of your device.

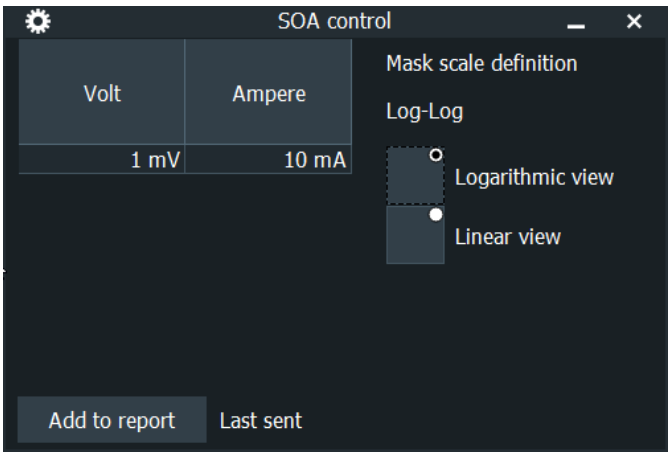
Required probes:

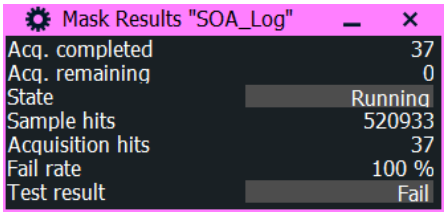
- Differential voltage probe
- Current probe

19.5.1.1 Safe operating area results

The results of " Safe Operating Area" measurements are provided in the following ways:

- The diagram shows the graphical presentation of:
  - the voltage waveform
  - the current waveform
- A logarithmic or linear XY diagram of the calculated voltage (x-axis) and current (y-axis) waveforms. This curve is a graphical representation of the power handling capability of the device under various conditions.
- The result box displays the numeric measurement results. Also, you can see the mask definition and change the scale in the "SOA Control" dialog. If the state of "Enable mask test" is "On" an extra result box appears, see also Table 19-3.





To measure and display the safe operating area, the instrument uses the following measurements and diagrams:

- "XY1" XY Diagram 1 to measure the logarithmic waveform
- "XY2" XY Diagram 2 to measure the linear waveform
- "M1" Math 1 to calculate the voltage signal
- "M2" Math 2 to calculate the current signal

The used resources are listed in the "Details" tab. See also: Chapter 19.2.3, "Details tab", on page 1135.

The results of the safe operating area mask test are described in Table 19-3.

Table 19-3: Results of the mask test

Result	Description
Acq. completed	Number of tested acquisitions
Acq. remaining	Remaining acquisitions until "Average count / Nx Single count" is reached
State	Shows if the test has been completed. The state is set to "Finished" when "Nx Single count" acquisitions are tested and the number of "Acq. remaining" is 0. As long as the number of tested acquisitions is less the "Nx Single count" number, the state is "Running". If you run the acquisition with [Run Stop], or the number of played history acquisitions exceeds "Nx Single count", the mask testing is performed according to fail criteria settings independently of the test state. The testing is not stopped when the state is set to "Finished".
Sample hits	Number of samples that hit the mask
Acquisition hits	Number of acquisitions that contained at least one sample hit
Fail rate	Ratio of acquisition hits to the number of tested acquisitions
Test result	A test has failed if the number of sample hits or acquisition hits exceeds the limit of "Violation tolerance" hits

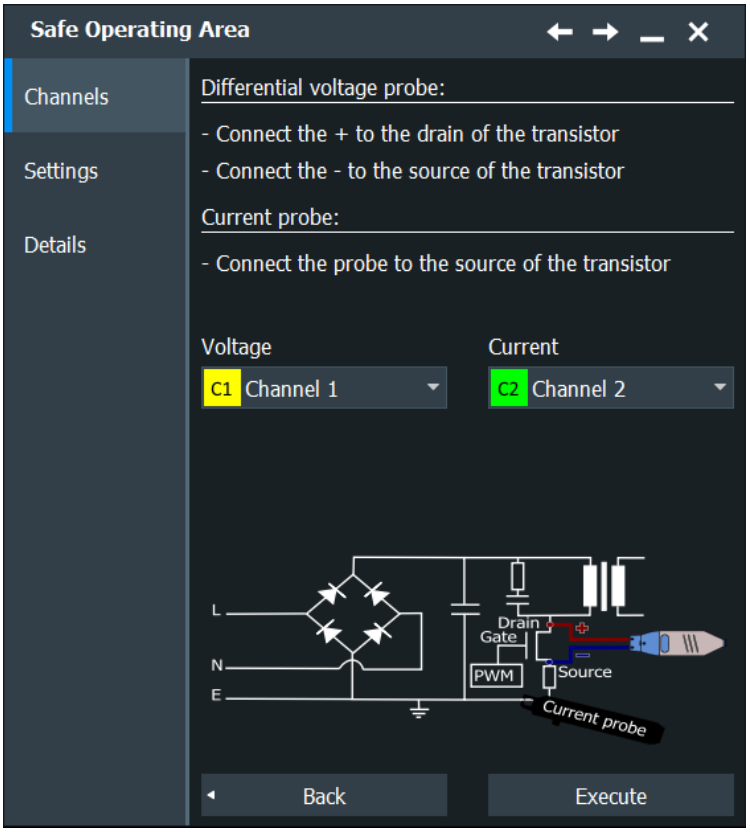
The following remote commands are used for handling the measurement results:

- `POWer:SOA:SWITch` on page 2541
- `POWer:SOA:REPort:ADD` on page 2541

19.5.1.2 Configuring safe operating area

For details of the configuration settings, see Chapter 19.5.1.3, "Safe operating area settings", on page 1167.

1. Select "Analysis" > "Power".
2. Under "Power Path", select "Safe Operating Area".
3. Connect the differential voltage probe and the current probe to the oscilloscope.
4. Deskew the probes as described in Chapter 19.1.1.1, "Auto deskew", on page 1123.
5. Connect the probes to the DUT as shown in the "Channels" tab:



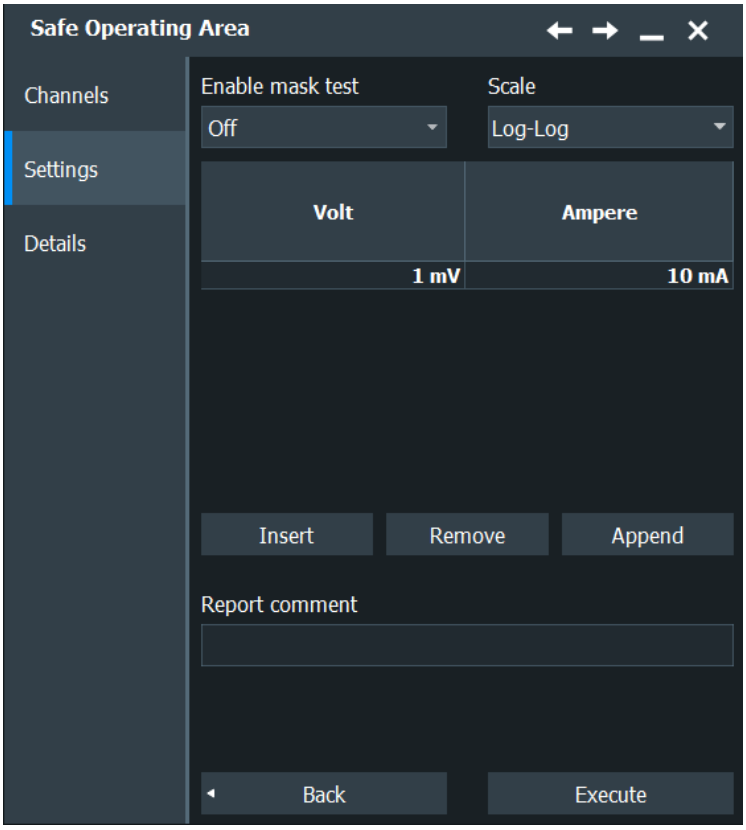
6. Select the correct channels of the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.
8. Select the state of the "Enable mask test".
9. Select the "Scale".
10. Define the SOA Points.
11. Tap "Execute".

    On the screen you can see the measurement waveforms of the Additionally, the result box with numeric measurement results is shown. For details, seeChapter 19.5.1.1, "Safe operating area results", on page 1165.

19.5.1.3 Safe operating area settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 19.2.1, "Channels tab", on page 1134.

In the "Settings" tab, you configure the safe operating area parameters and display settings.



Enable mask test
Enables a mask test.

Remote command:
`POWer:SOA:MASK` on page 2541

Scale
Selects the scale for the result diagram.

Remote command:
`POWer:SOA:SCALe` on page 2541

SOA point definition
In this table, you can set voltage-current points to define a mask for the safe point operating area. If Enable mask test > "On", you can check whether the signal remains within the specified limits.

To add a point press "Insert" or "Append". To remove a point press "Remove".

Remote command:
`POWer:SOA:LINear:ADD` on page 2539
`POWer:SOA:LOGarithmic:ADD` on page 2539
`POWer:SOA:LINear:COUNt?` on page 2540
`POWer:SOA:LOGarithmic:COUNt?` on page 2540
`POWer:SOA:LINear:INSert` on page 2540
`POWer:SOA:LOGarithmic:INSert` on page 2540

POWer:SOA:LINear:REMove on page 2540
POWer:SOA:LOGarithmic:REMove on page 2540

Volt ← SOA point definition
Sets the voltage value of the SOA point.

Remote command:
POWer:SOA:LINear:POINt<m>:VOLTage on page 2540

Ampere ← SOA point definition
Sets the current value of the SOA point.

Remote command:
POWer:SOA:LINear:POINt<m>:CURRent on page 2540

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Safe Operating Area" measurement.

Remote command:
POWer:SOA:EXECute on page 2539

19.5.2 Turn on/off

"Turn On/Off" analysis measures the time that a power supply needs to reach a certain percentage of the steady state output level when initially turned on or turned off.

Common measuring scenarios include:

- Turn on time: measurement of the time it takes for the DC output to reach 90 % of the expected steady state level, after the power supply is initially turned on.
- Turn off time: measurement of the time it takes for the DC output to reach 10 % of the expected steady state level, after the power supply is initially turned off.

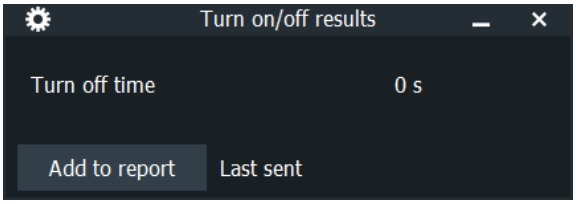
Required probes:

- Differential voltage probe
- Passive or differential voltage probe

19.5.2.1 Turn on/off results

The results of "Turn on/off" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  - the input voltage waveform
  - the output voltage waveform
- The result box displays the "Turn on time" or the "Turn off time".



The used resources are listed in the "Details" tab. See also: Chapter 19.2.3, "Details tab", on page 1135.

The" Turn on time" is measured as the time between the trigger point ("Trigger level on" value is reached) and the time the given percentage of the "Steady state level" is reached.

The" Turn off time" is measured as the time between the trigger point, delayed with the set "Time" ("Trigger level on" value is reached) and the time the given percentage of the "Steady state level" is reached, see

See also Figure 19-2 and Figure 19-3

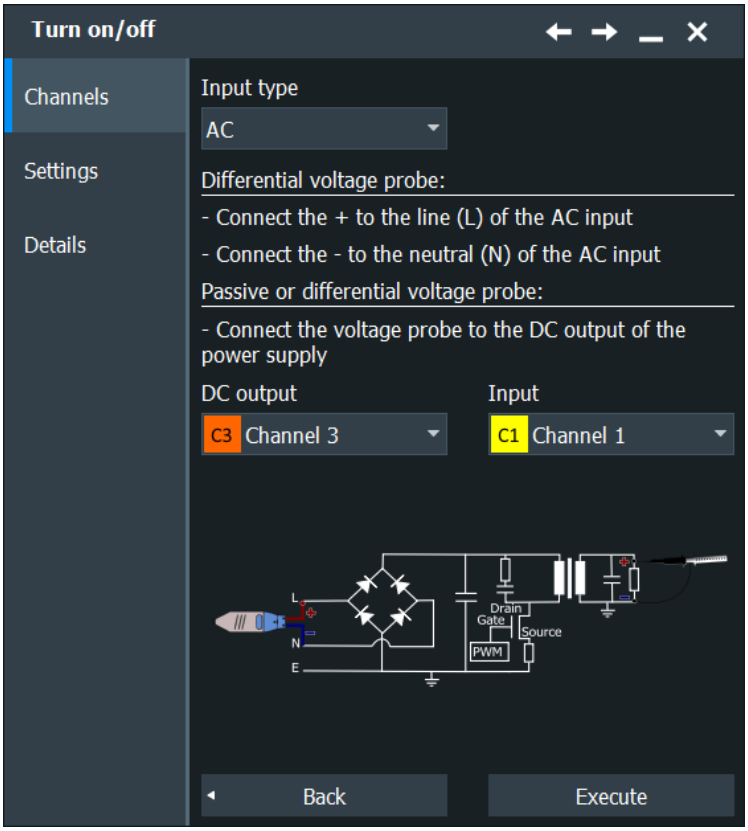
The following remote commands are used for handling the measurement results:

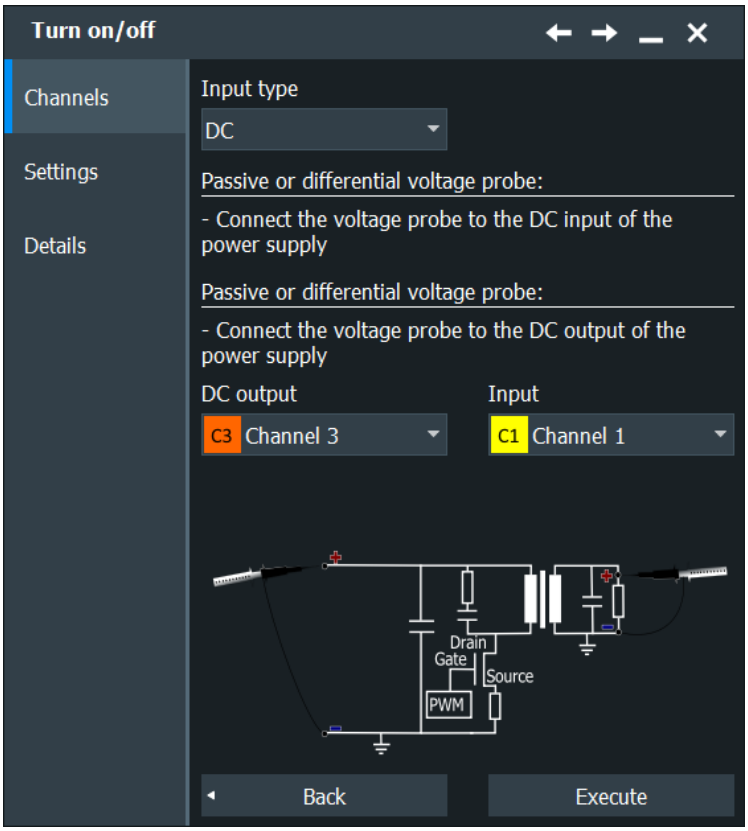
- `POWer:ONOFf:RESult:TOFF?` on page 2543
- `POWer:ONOFf:RESult:TON?` on page 2543
- `POWer:ONOFf:REPort:ADD` on page 2542

19.5.2.2 Configuring turn on/ off

For details of the configuration settings, see Chapter 19.5.2.3, "Turn on/off settings", on page 1172.

1. Select "Analysis">"Power".
2. Under "Power Path", select "Turn on/off ".
3. Connect the probes to the DUT as shown in the "Channels" tab:


Turn on/off
Channels
Settings
Details
Input type
AC
Differential voltage probe:
- Connect the + to the line (L) of the AC input
- Connect the - to the neutral (N) of the AC input
Passive or differential voltage probe:
- Connect the voltage probe to the DC output of the power supply
DC output
Input
C3 Channel 3
C1 Channel 1
L
N
E
Drain
Gate
Source
PWM
Back
Execute


Turn on/off
Channels
Settings
Details
Input type
DC
Passive or differential voltage probe:
- Connect the voltage probe to the DC input of the power supply
Passive or differential voltage probe:
- Connect the voltage probe to the DC output of the power supply
DC output
Input
C3 Channel 3
C1 Channel 1
Drain
Gate
Source
PWM
Back
Execute

4. Select the correct channels for the "DC output" and the "AC input" or the "DC input".
5. Select the "Settings" tab.
6. Select whether you want to measure "Turn on" or "Turn off".
7. Set the "Steady state level" and the "Trigger level " according to your requirements.
8. Tap "Execute".
9. Turn on/off the DUT.

   On the screen, you can see the measurement. Also, the result box displays the turn on or the turn off time. For details, see Chapter 19.5.2.1, "Turn on/off results", on page 1169.

19.5.2.3 Turn on/off settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 19.2.1, "Channels tab", on page 1134.

In the "Settings" tab, you configure the turn on and the turn off parameters.

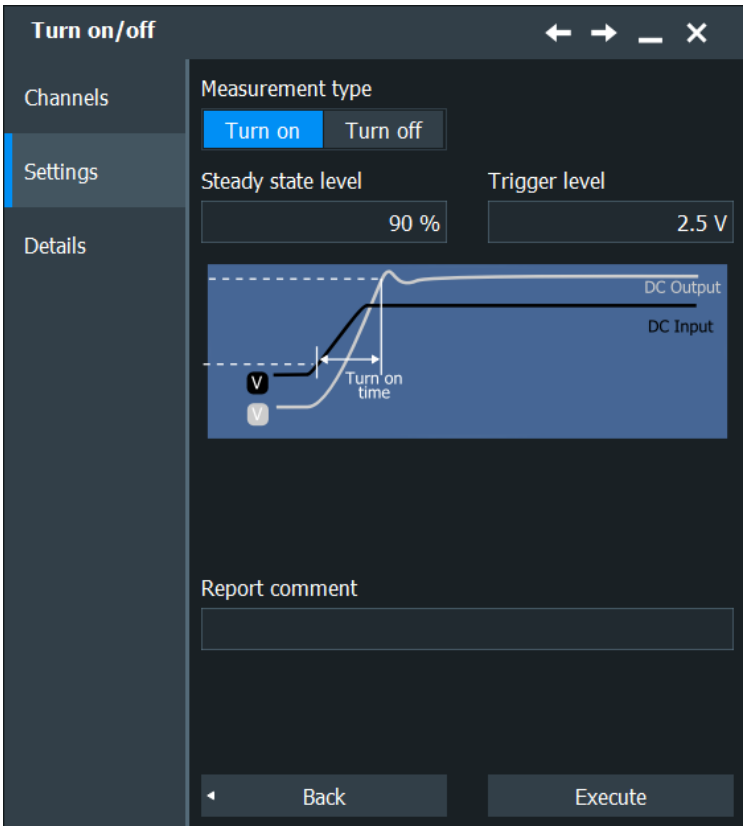


Figure 19-2: Settings turn on time

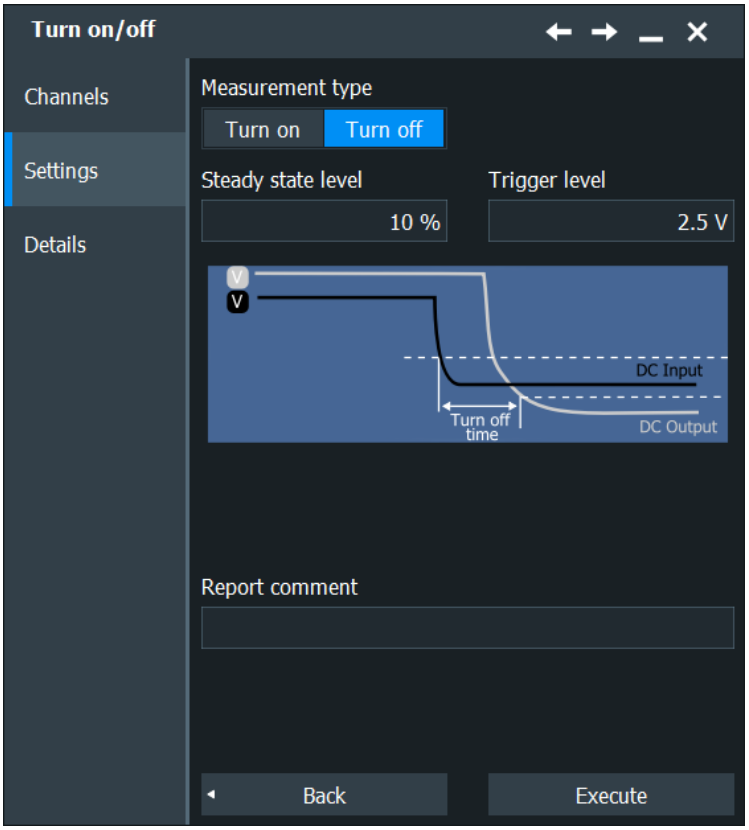


Figure 19-3: Settings turn off time

Input Type
To access this setting, select the "Channels" tab.

Selects the AC or DC input type.

Remote command:
`POWer:ONOFf:INPut` on page 2542

Measurement Type
Selects the"Turn on" or the " Turn off " measurement.

Remote command:
`POWer:ONOFf:TYPE` on page 2543

Turn on
Enables the configuration of the turn on time measurement.

Steady state level-Turn on ← Turn on
Sets the percentage of the steady state level of the DC output that has to be reached.

Remote command:
`POWer:ONOFf:DSON` on page 2542

Trigger level on ← Turn on
Triggers the beginning of the measurements at the moment the AC or DC input voltage reaches the set value.

Remote command:
`POWer:ONOFf:ATON` on page 2542
`POWer:ONOFf:DTON` on page 2542

Turn off
Enables the configuration of the turn off time measurement.

Steady state level- Turn off ← Turn off
Sets the percentage of the steady state level of the DC output that has to be reached.

Remote command:
`POWer:ONOFf:DSOFf` on page 2542

Trigger level ← Turn off
Triggers the beginning of the measurements at the moment the AC or DC input voltage reaches the set value.

Remote command:
`POWer:ONOFf:ATOFf` on page 2542
`POWer:ONOFf:DTOFf` on page 2542

Time ← Turn off
Sets the time, the start of the measurement of the turn off time is delayed with, after the trigger point.

Remote command:
`POWer:ONOFf:TIME` on page 2543

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the " Power Quality" measurement.

Remote command:
`POWer:ONOFf:EXECute` on page 2542

19.5.3 Switching loss

The "Switching Loss" analysis measures the power and energy losses of a switching device, that occur during the switching phases and the conduction phase of the switching transistor.

Required probes:

- Differential voltage probe
- Current probe

19.5.3.1 Switching loss results

The results of "Switching Loss" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  - the voltage waveform
  - the current waveform
  - the power waveform
- The result box displays the numeric measurement results for the enabled measurement parameters in dependence of the energy and the power. To switch the display, tap "Energy" or "Power" accordingly.
  For a detailed description, see Table 19-2.

Switching Loss Results

	Current	+Peak	-Peak	μ (Avg)	RMS	σ (S-dev)	Event count	Wave count
Switching frequency	0 Hz	0 Hz	0 Hz	0 Hz	0 Hz	0 Hz	0	0
Turn on	0 W	0 W	0 W	0 W	0 W	0 W	0	0
Turn off	0 W	0 W	0 W	0 W	0 W	0 W	0	0
Conduction	0 W	0 W	0 W	0 W	0 W	0 W	0	0
Non conduction	0 W	0 W	0 W	0 W	0 W	0 W	0	0
Total	0 W	0 W	0 W	0 W	0 W	0 W	0	0

Reset statistics | Add to report | Last sent | Energy >

To measure and display the switching loss, the instrument uses the following measurements and cursors:

- "P3"... "P8": "Meas 3" ... "Meas 8" to measure the voltage
- "M4" Math 4 to calculate the power
- "C3" Cursor 3 to define time points "t_0" and "t_1"
- "C4" Cursor 4 to define time points "t_2" and "t_3"

The used resources are listed in the "Details" tab. See also: Chapter 19.2.3, "Details tab", on page 1135.

"Switching Loss" is a statistical evaluation that is reset only if the measurement setup is changed or you reset the statistics.

The switching loss phases that can be defined during the measurement are shown in Figure 19-4 and described in Table 19-4.

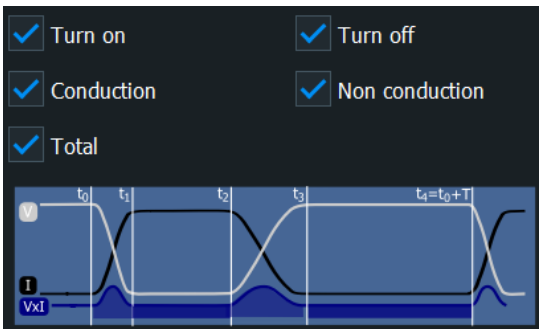


Figure 19-4: Switching loss phases

Table 19-4: Switching loss phases

Phase	Definition Points	Description
Turn on	The area between "t_0" and "t_1"	The time after switching the device, during which the current rises until it reaches the saturation current level.
Conduction	The area between "t_1" and "t_2"	The time during which the voltage is at the transistors saturated minimum and the current flows.
Turn off	The area between "t_2" and "t_3"	The time during which after a short delay time the voltage rises until it reaches its final value.
Non conduction	The area between "t_3" and "t_4"	The time during current does not flow. The losses during this period should be theoretically zero.
Total	The area between "t_0" and "t_4"	The period of one switching cycle.

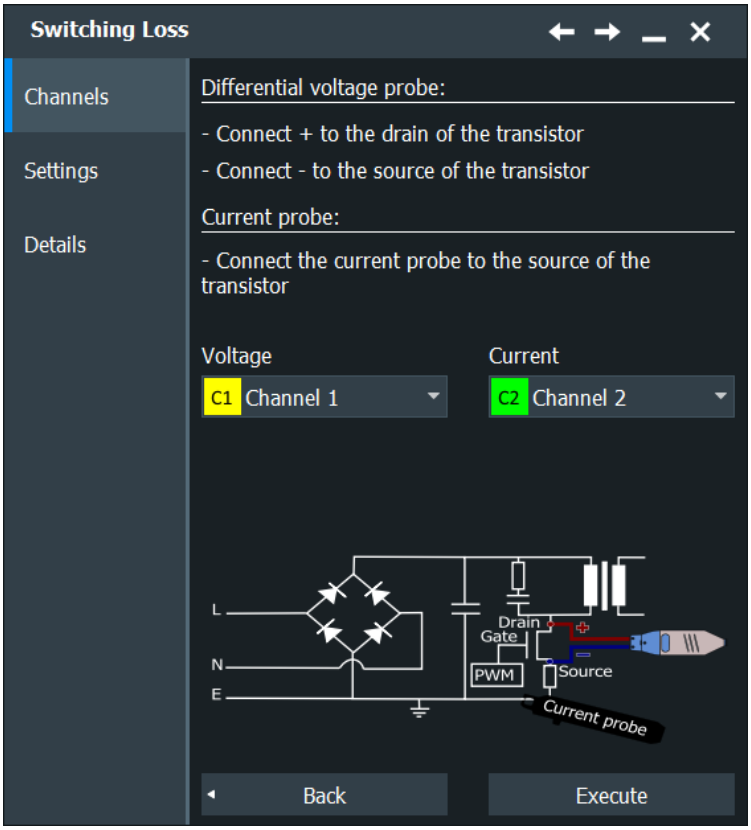
The following remote commands are used for handling the measurement results:

- `POWer:SWITching:GATE:COND:STARt` on page 2545
- `POWer:SWITching:GATE:COND:STOP` on page 2545
- `POWer:SWITching:GATE:NCON:STARt` on page 2545
- `POWer:SWITching:GATE:TOFF:STARt` on page 2545
- `POWer:SWITching:GATE:TOFF:STOP` on page 2545
- `POWer:SWITching:GATE:TON:STARt` on page 2545
- `POWer:SWITching:GATE:TON:STOP` on page 2545
- `POWer:SWITching:RESult:ENERgy:ACTual?` on page 2545
- `POWer:SWITching:RESult:ENERgy:AVG?` on page 2545
- `POWer:SWITching:RESult:ENERgy:EVTCount?` on page 2545
- `POWer:SWITching:RESult:ENERgy:NPEak?` on page 2545
- `POWer:SWITching:RESult:ENERgy:PPEak?` on page 2545
- `POWer:SWITching:RESult:ENERgy:RMS?` on page 2545
- `POWer:SWITching:RESult:ENERgy:STDDev?` on page 2545
- `POWer:SWITching:RESult:ENERgy:WFMCount?` on page 2545
- `POWer:SWITching:RESult:POWer:ACTual?` on page 2546
- `POWer:SWITching:RESult:POWer:AVG?` on page 2546
- `POWer:SWITching:RESult:POWer:EVTCount?` on page 2546
- `POWer:SWITching:RESult:POWer:NPEak?` on page 2546
- `POWer:SWITching:RESult:POWer:PPEak?` on page 2546
- `POWer:SWITching:RESult:POWer:RMS?` on page 2546
- `POWer:SWITching:RESult:POWer:STDDev?` on page 2546
- `POWer:SWITching:RESult:POWer:WFMCount?` on page 2546
- `POWer:SWITching:REPort:ADD` on page 2544

19.5.3.2 Configuring switching loss

For details of the configuration settings, see Chapter 19.5.3.3, "Switching loss settings", on page 1178.

1. Select "Analysis" > "Power".
2. Under "Power Path", select "Switching Loss".
3. Connect the differential voltage probe and the current probe to the oscilloscope.
4. Deskew the probes as described in Chapter 19.1.1.1, "Auto deskew", on page 1123.
5. Connect the probes to the DUT as shown in the "Channels" tab:



6. Select the correct channels for the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.
8. Enable the parameters you want to measure.
9. Select an"Optimize Scaling" option.

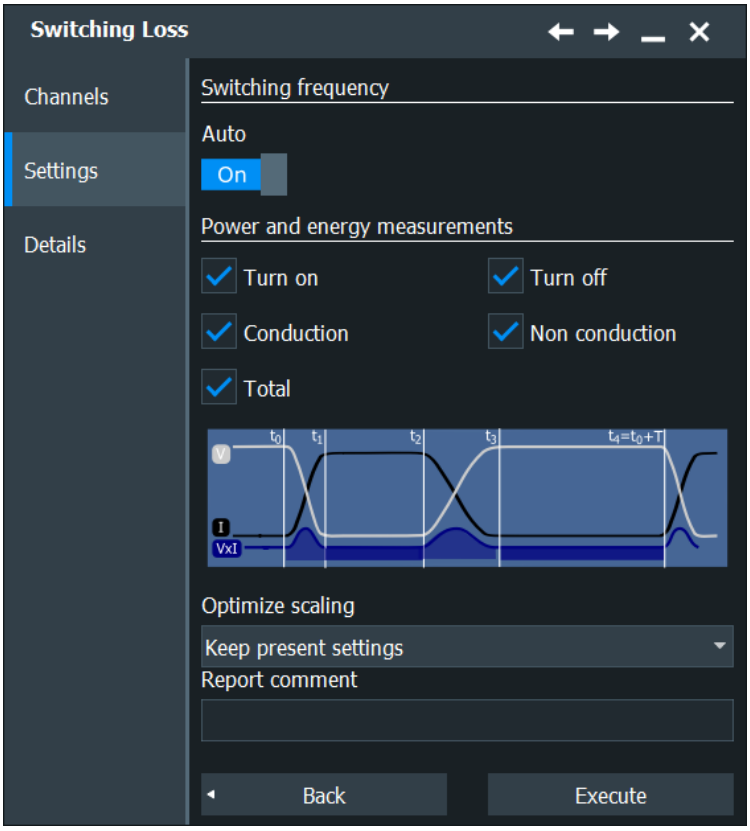
10. Tap "Execute".

    On the screen you can see the measurement waveforms of the current, the voltage and the power. Also, the result box with numeric measurement results is shown. For details, see Chapter 19.5.3.1, "Switching loss results", on page 1175.

19.5.3.3 Switching loss settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 19.2.1, "Channels tab", on page 1134.

In the "Settings" tab, you configure the switching loss parameters and display settings.



Switching frequency > Auto
Enables the measurements of the switching frequency. If disabled you can enter the value of the switching frequency.

Remote command:
`POWer:SWITching:SWIFrequency` on page 2544
`POWer:SWITching:SWIT` on page 2545

Turn on
Enables the measurements during the turn on period.

Remote command:
`POWer:SWITching:TON` on page 2545

Conduction
Enables the measurements during the conduction period.

Remote command:
`POWer:SWITching:COND` on page 2545

Turn off
Enables the measurements during the turn off period.

Remote command:
`POWer:SWITching:TOFF` on page 2545

Non conduction
Enables the measurements during the non conduction period.

Remote command:
`POWer:SWITching:NCON` on page 2545

Total
Enables the measurements of the total period

Remote command:
`POWer:SWITching:TOTal` on page 2545

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:SWITching:AUTO` on page 2544

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the " Switching Loss" measurement.

Remote command:
`POWer:SWITching:EXECute` on page 2544

19.5.4 Power efficiency

This measurement requires a 4-channel oscilloscope (R&S RTO6xxx4).

"Power Efficiency" analysis measures the input and the output power of a power supply. The power efficiency of the power supply is then calculated as the ratio of the output power and the input power.

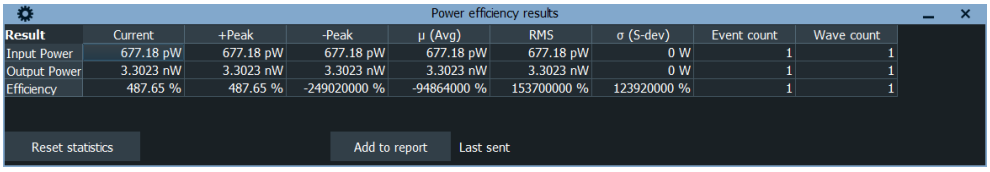
Required probes:

- Two differential voltage probes
- Two current probes

19.5.4.1 Power efficiency results

The results of "Power Efficiency" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  - the voltage input waveform
  - the current input waveform
  - the voltage output waveform
  - the voltage input waveform
  - the power input waveform
  - the power output waveform
- The result box displays the numeric measurement results of the "Input power", "Output power" and the "Efficiency". For a detailed description, see Table 19-2.


Power efficiency results

Result	Current	+Peak	-Peak	μ (Avg)	RMS	σ (S-dev)	Event count	Wave count
Input Power	677.18 pW	677.18 pW	677.18 pW	677.18 pW	677.18 pW	0 W	1	1
Output Power	3.3023 nW	3.3023 nW	3.3023 nW	3.3023 nW	3.3023 nW	0 W	1	1
Efficiency	487.65 %	487.65 %	-249020000 %	-94864000 %	153700000 %	123920000 %	1	1

Reset statistics Add to report Last sent

To measure and display the power quality, the instrument uses the following measurements and waveforms:

- "MG 7" to measure the input power waveform
- "MG 8" to measure the output power waveform
- "M2" Math 2 to calculate the input power
- "M3" Math 3 to calculate the output power

The used resources are listed in the "Details" tab. See also: Chapter 19.2.3, "Details tab", on page 1135.

"Power Efficiency" is a statistical evaluation that is reset only if the measurement setup is changed or you reset the statistics.

The following remote commands are used for handling the measurement results:

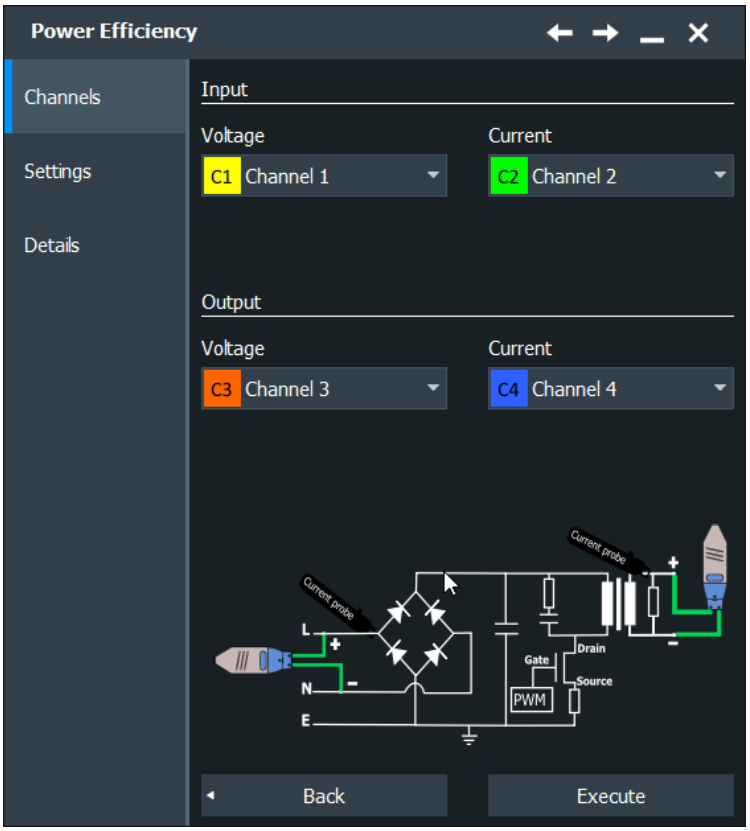
- `POWer:EFFiciency:RESult<m>:ACTual?` on page 2547
- `POWer:EFFiciency:RESult<m>:AVG?` on page 2547
- `POWer:EFFiciency:RESult<m>:EVTCount?` on page 2547
- `POWer:EFFiciency:RESult<m>:NPEak?` on page 2547
- `POWer:EFFiciency:RESult<m>:PPEak?` on page 2547
- `POWer:EFFiciency:RESult<m>:RMS?` on page 2547
- `POWer:EFFiciency:RESult<m>:STDDev?` on page 2547
- `POWer:EFFiciency:RESult<m>:WFMCount?` on page 2547

- `POWer:EFFiciency:REPort:ADD` on page 2547

19.5.4.2 Configuring power efficiency

For details of the configuration settings, see Chapter 19.5.4.3, "Power efficiency settings", on page 1182.

1. Select "Analysis" > "Power".
2. Under "Power Path", select "Power Efficiency".
3. Connect the differential voltage probes and the current probes to the oscilloscope.
4. Deskew the probes as described in Chapter 19.1.1.1, "Auto deskew", on page 1123.
5. Connect the probes to the DUT as shown in the graphic of the "Channels" tab:



6. Select the correct channels for the "Current Source" and the "Voltage Source" of the input and the output.
7. Select the "Settings" tab.
8. Select an "Optimize Scaling" option.

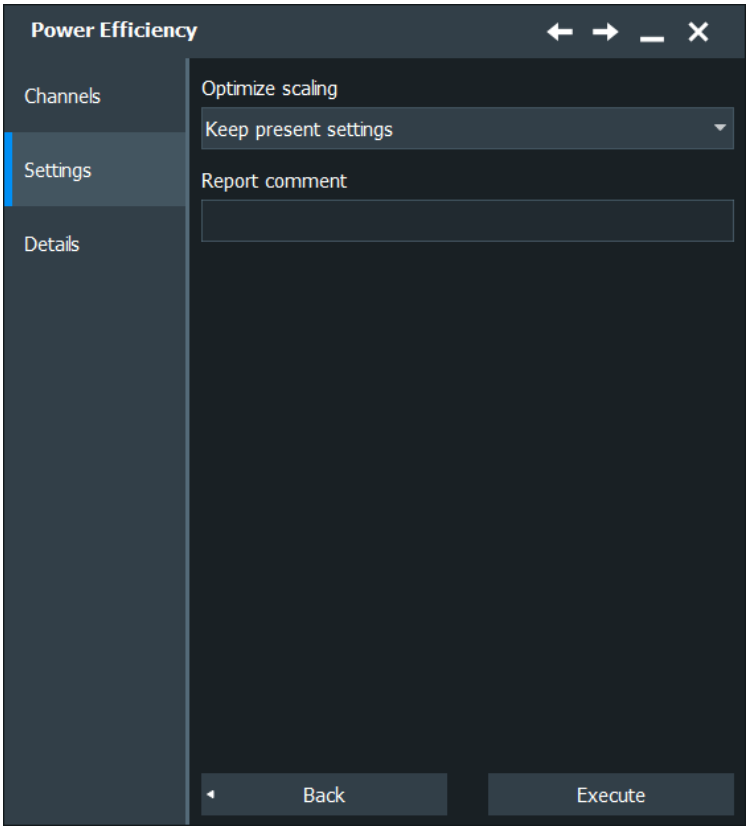
9. Tap "Execute".

   On the screen, you can see the measurement waveforms of the input power and the output power. Also, the result box with numeric measurement results is shown. For details, see Chapter 19.5.4.1, "Power efficiency results", on page 1180

19.5.4.3 Power efficiency settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 19.2.1, "Channels tab", on page 1134.

In the "Settings" tab, you configure the power efficiency display settings.



Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:EFFiciency:AUTO` on page 2546

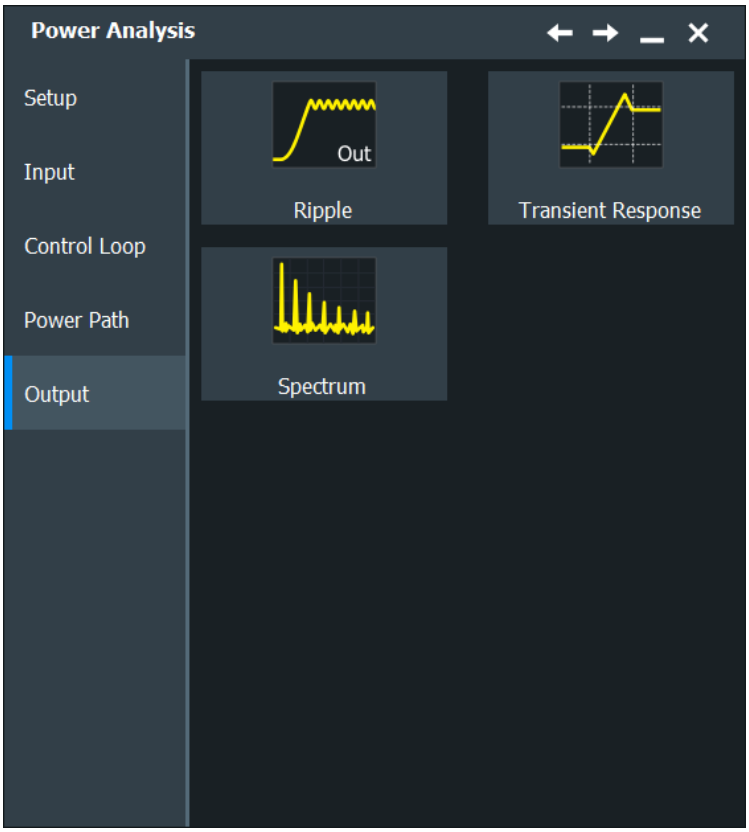
Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the " Power Effieciency" measurement.

Remote command:
`POWer:EFFiciency:EXECute` on page 2547

19.6 Output

Access: [App Cockpit]> "Analysis" > "Power" > "Output"



The "Output" measurements are used for characterizing the behavior and quality of the SMPS output voltage.

19.6.1 Output ripple

The "Output Ripple" analysis measures the ripple of the device output. You can measure the voltage ripple alone or the voltage and the current ripple simultaneously. In this measurement, the peak-to peak extremes of the output DC signal are of interest.

The measurement also includes the AC-RMS of the output DC signal, that is calculated as a standard derivation.

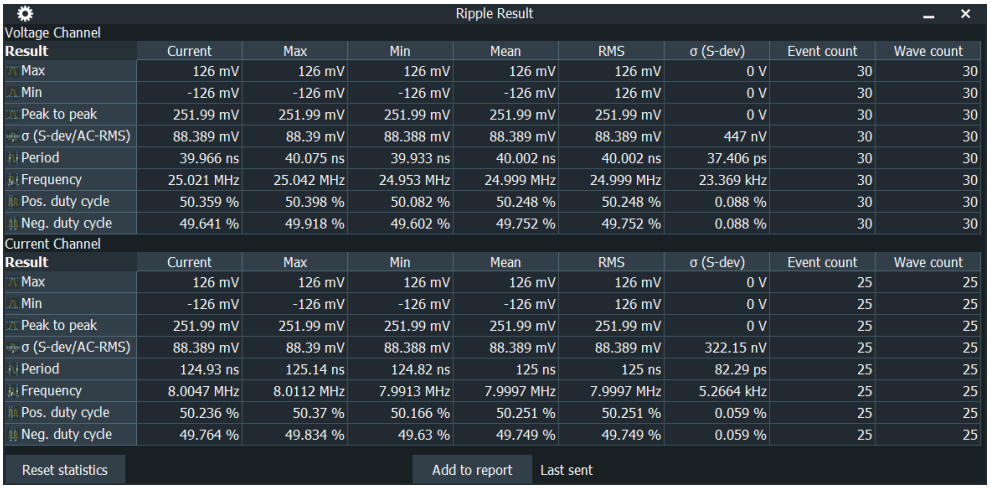
Required probes:

- Voltage probe
- (Optional) Current probe

19.6.1.1 Output ripple results

The results of "Output Ripple" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  - the voltage waveform
  - (optional) the current waveform
- The result box displays the numeric measurement results for the voltage and for the current ripple. For details, see Table 19-5 and Table 19-2.



Ripple Result

Voltage Channel

Result	Current	Max	Min	Mean	RMS	σ (S-dev)	Event count	Wave count
Max	126 mV	126 mV	126 mV	126 mV	126 mV	0 V	30	30
Min	-126 mV	-126 mV	-126 mV	-126 mV	126 mV	0 V	30	30
Peak to peak	251.99 mV	251.99 mV	251.99 mV	251.99 mV	251.99 mV	0 V	30	30
σ (S-dev/AC-RMS)	88.389 mV	88.39 mV	88.388 mV	88.389 mV	88.389 mV	447 nV	30	30
Period	39.966 ns	40.075 ns	39.933 ns	40.002 ns	40.002 ns	37.406 ps	30	30
Frequency	25.021 MHz	25.042 MHz	24.953 MHz	24.999 MHz	24.999 MHz	23.369 kHz	30	30
Pos. duty cycle	50.359 %	50.398 %	50.082 %	50.248 %	50.248 %	0.088 %	30	30
Neg. duty cycle	49.641 %	49.918 %	49.602 %	49.752 %	49.752 %	0.088 %	30	30

Current Channel

Result	Current	Max	Min	Mean	RMS	σ (S-dev)	Event count	Wave count
Max	126 mV	126 mV	126 mV	126 mV	126 mV	0 V	25	25
Min	-126 mV	-126 mV	-126 mV	-126 mV	126 mV	0 V	25	25
Peak to peak	251.99 mV	251.99 mV	251.99 mV	251.99 mV	251.99 mV	0 V	25	25
σ (S-dev/AC-RMS)	88.389 mV	88.39 mV	88.388 mV	88.389 mV	88.389 mV	322.15 nV	25	25
Period	124.93 ns	125.14 ns	124.82 ns	125 ns	125 ns	82.29 ps	25	25
Frequency	8.0047 MHz	8.0112 MHz	7.9913 MHz	7.9997 MHz	7.9997 MHz	5.2664 kHz	25	25
Pos. duty cycle	50.236 %	50.37 %	50.166 %	50.251 %	50.251 %	0.059 %	25	25
Neg. duty cycle	49.764 %	49.834 %	49.63 %	49.749 %	49.749 %	0.059 %	25	25

Reset statistics Add to report Last sent

To measure and display the output ripple, the instrument uses the following measurements and waveforms:

- "MG 7" to measure the current
- "MG 8" to measure the voltage

The used resources are listed in the "Details" tab. See also: Chapter 19.2.3, "Details tab", on page 1135.

Table 19-5: Properties output ripple

	Meas. type	Symbol	Description/Result
	Max	X_{Max}	Absolute maximum value of the waveform.
	Min	X_{Min}	Absolute minimum value of the waveform.

	Meas. type	Symbol	Description/Result
	Peak to peak	X_{PkPk}	Peak-to-peak value of the waveform: the difference of maximum and minimum values. $X_{Ampl} = X_{Max} - X_{Min}$
	σ (S-dev/ AC-RMS)	σ_X	Standard deviation of the waveform samples
	Period	T_{Period}	Time of the left-most signal period of the waveform - the time difference between two consecutive waveform edges measured on the middle reference level. The measurement requires at least one complete period of a triggered signal. Multiple measurements are possible.
	Frequency	f_{Period}	Frequency of the signal, reciprocal value of the period. $f_{Period} = 1 / T_{Period}$
	Pos. duty cycle	R_{PosCyc}	Positive duty cycle: Width of a positive pulse in relation to the period in %. The measurement requires at least one complete period of a triggered signal. Multiple measurements are possible. $R_{PosCyc} = \frac{T_{PosPulse}}{T_{Period}} \cdot 100\%$
	Neg. duty cycle	R_{NegCyc}	Negative duty cycle: Width of a negative pulse in relation to the period in %. The measurement requires at least one complete period of a triggered signal. Multiple measurements are possible. $R_{NegCyc} = \frac{T_{NegPulse}}{T_{Period}} \cdot 100\%$

"Ripple" is a statistical evaluation that is reset only if the measurement setup is changed or you reset the statistics.

The following remote commands are used for handling the measurement results:

- `POWer:RIPPle:RESult:FREQuency:AVG?` on page 2550
- `POWer:RIPPle:RESult:FREQuency:EVTCount?` on page 2550
- `POWer:RIPPle:RESult:FREQuency:NPEak?` on page 2550
- `POWer:RIPPle:RESult:FREQuency:PPEak?` on page 2550
- `POWer:RIPPle:RESult:FREQuency:RMS?` on page 2550
- `POWer:RIPPle:RESult:FREQuency:STDDev?` on page 2550
- `POWer:RIPPle:RESult:FREQuency:WFMCount?` on page 2550
- `POWer:RIPPle:RESult:FREQuency[:ACTual]?` on page 2550
- `POWer:RIPPle:RESult:MAXimum:AVG?` on page 2550
- `POWer:RIPPle:RESult:MAXimum:EVTCount?` on page 2550
- `POWer:RIPPle:RESult:MAXimum:NPEak?` on page 2550

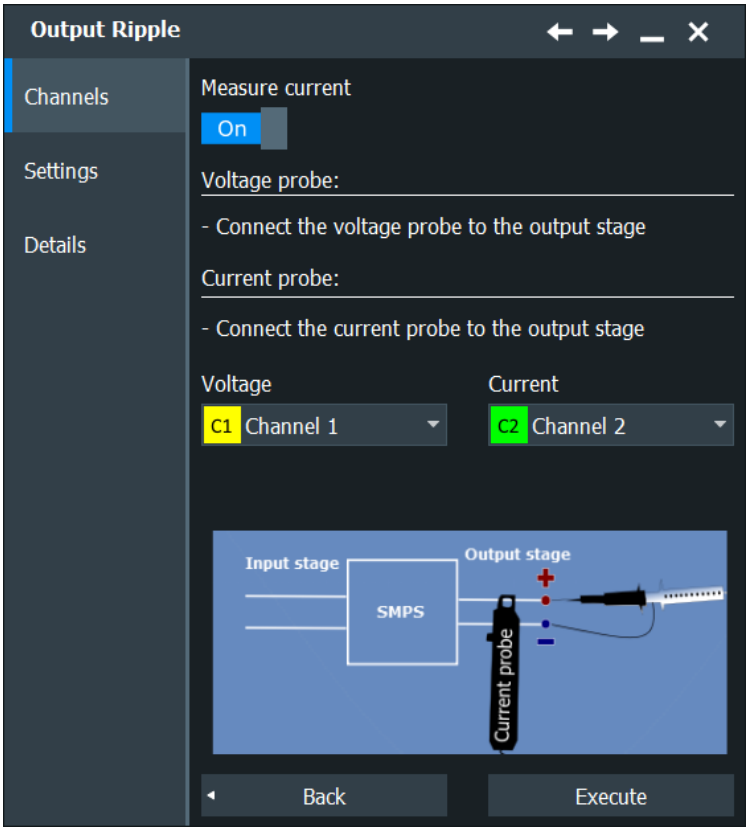
- POWer:RIPPle:RESult:MAXimum:PPEak? on page 2550
- POWer:RIPPle:RESult:MAXimum:RMS? on page 2550
- POWer:RIPPle:RESult:MAXimum:STDDev? on page 2550
- POWer:RIPPle:RESult:MAXimum:WFMCount? on page 2550
- POWer:RIPPle:RESult:MAXimum[:ACTual]? on page 2550
- POWer:RIPPle:RESult:MINimum:AVG? on page 2551
- POWer:RIPPle:RESult:MINimum:EVTCount? on page 2551
- POWer:RIPPle:RESult:MINimum:NPEak? on page 2551
- POWer:RIPPle:RESult:MINimum:PPEak? on page 2551
- POWer:RIPPle:RESult:MINimum:RMS? on page 2551
- POWer:RIPPle:RESult:MINimum:STDDev? on page 2551
- POWer:RIPPle:RESult:MINimum:WFMCount? on page 2551
- POWer:RIPPle:RESult:MINimum[:ACTual]? on page 2551
- POWer:RIPPle:RESult:NDCYcle:AVG? on page 2551
- POWer:RIPPle:RESult:NDCYcle:EVTCount? on page 2551
- POWer:RIPPle:RESult:NDCYcle:NPEak? on page 2551
- POWer:RIPPle:RESult:NDCYcle:PPEak? on page 2551
- POWer:RIPPle:RESult:NDCYcle:RMS? on page 2551
- POWer:RIPPle:RESult:NDCYcle:STDDev? on page 2552
- POWer:RIPPle:RESult:NDCYcle:WFMCount? on page 2552
- POWer:RIPPle:RESult:NDCYcle[:ACTual]? on page 2551
- POWer:RIPPle:RESult:PDCYcle:AVG? on page 2552
- POWer:RIPPle:RESult:PDCYcle:EVTCount? on page 2552
- POWer:RIPPle:RESult:PDCYcle:NPEak? on page 2552
- POWer:RIPPle:RESult:PDCYcle:PPEak? on page 2552
- POWer:RIPPle:RESult:PDCYcle:RMS? on page 2552
- POWer:RIPPle:RESult:PDCYcle:STDDev? on page 2552
- POWer:RIPPle:RESult:PDCYcle:WFMCount? on page 2552
- POWer:RIPPle:RESult:PDCYcle[:ACTual]? on page 2552
- POWer:RIPPle:RESult:PDEL:AVG? on page 2553
- POWer:RIPPle:RESult:PDEL:EVTCount? on page 2553
- POWer:RIPPle:RESult:PDEL:NPEak? on page 2553
- POWer:RIPPle:RESult:PDEL:PPEak? on page 2553
- POWer:RIPPle:RESult:PDEL:RMS? on page 2553
- POWer:RIPPle:RESult:PDEL:STDDev? on page 2553
- POWer:RIPPle:RESult:PDEL:WFMCount? on page 2553
- POWer:RIPPle:RESult:PDEL[:ACTual]? on page 2553
- POWer:RIPPle:RESult:PERiod:AVG? on page 2553

- POWer:RIPPle:RESult:PERiod:EVTCount? on page 2553
- POWer:RIPPle:RESult:PERiod:NPEak? on page 2553
- POWer:RIPPle:RESult:PERiod:PPEak? on page 2553
- POWer:RIPPle:RESult:PERiod:RMS? on page 2553
- POWer:RIPPle:RESult:PERiod:STDDev? on page 2553
- POWer:RIPPle:RESult:PERiod:WFMCount? on page 2553
- POWer:RIPPle:RESult:PERiod[:ACTual]? on page 2553
- POWer:RIPPle:RESult:STDDev:AVG? on page 2554
- POWer:RIPPle:RESult:STDDev:EVTCount? on page 2554
- POWer:RIPPle:RESult:STDDev:NPEak? on page 2554
- POWer:RIPPle:RESult:STDDev:PPEak? on page 2554
- POWer:RIPPle:RESult:STDDev:RMS? on page 2554
- POWer:RIPPle:RESult:STDDev:STDDev? on page 2554
- POWer:RIPPle:RESult:STDDev:WFMCount? on page 2554
- POWer:RIPPle:RESult:STDDev[:ACTual]? on page 2554
- POWer:RIPPle:REPort:ADD on page 2550

19.6.1.2 Configuring output ripple

For details of the configuration settings, see Chapter 19.6.1.3, "Output ripple settings", on page 1188.

1. Select "Analysis" > "Power".
2. Under "Output", select "Ripple".
3. Connect the voltage probe to the oscilloscope.
4. If you want to measure the current ripple, enable the current channel .Connect the current probe to the oscilloscope.
5. If you want to measure both the voltage and the current ripple, deskew the probes as described in Chapter 19.1.1.1, "Auto deskew", on page 1123.
6. Connect the probes to the DUT as shown in the "Channels" tab:



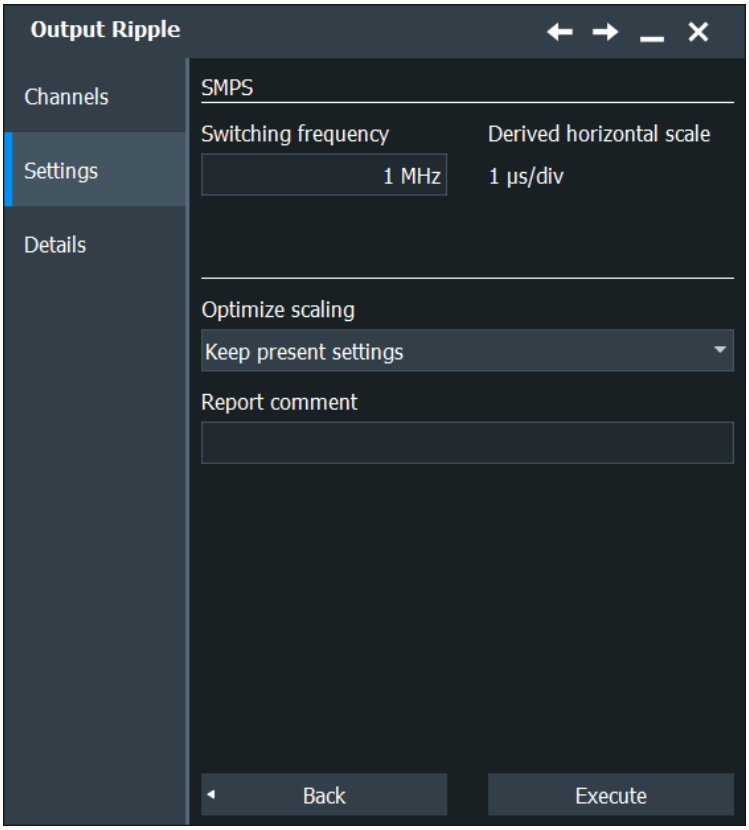
7. Select the correct channels for the "Voltage Source" and the "Current Source".
8. Select the "Settings" tab.
9. Set the "SMPS switching frequency" according to your signal.
10. Select an "Optimize Scaling" option.
11. Tap "Execute".

    On the screen, you can see the measurement waveforms of the current and the voltage. Also, the result box with numeric measurement results is shown. For details, see Chapter 19.6.1.1, "Output ripple results", on page 1184.

19.6.1.3 Output ripple settings

In the "Channels" tab, you set the current source and the voltage source, see also: Chapter 19.2.1, "Channels tab", on page 1134 and `POWer:RIPPle:CURRent` on page 2549)

In the "Settings" tab, you configure the ripple parameters and display settings.



Switching frequency
Sets the SMPS switching frequency. Set the value according to your DUT.

Remote command:
`POWer:RIPPle:FREQuency` on page 2549

Derived horizontal scale
Shows the scale of the displayed results. This value is calculated as the inverse of the "SMPS switching frequency"

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:RIPPle:AUToscale` on page 2549

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Output Ripple" measurement.

Remote command:
`POWer:RIPPle:EXECute` on page 2549

19.6.2 Transient response

The "Transient Response" analysis measures the response of a system to a change from equilibrium. This response is described by different properties like the rise time, the overshoot, the settling time, the peak time and the delay time.

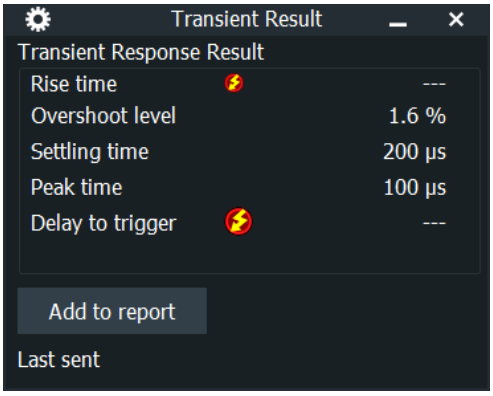
Required probes:

- One or two voltage probes
- Current probe

19.6.2.1 Transient response results

The results of " Power Quality" measurements are provided in two ways:

- The diagram shows the graphical presentation of:
  - the voltage waveforms
  - the current waveform
- The result box displays the numeric measurement results.



To measure and display the transient response, the instrument uses the following measurements and waveforms:

- MG 8 to measure the rise time, the overshoot and delay to trigger
- "Cu 1" Cursor 3 to measure the peak time
- "Cu 2" Cursor 2 to measure the delay to trigger

The used resources are listed in the "Details" tab. See also: Chapter 19.2.3, "Details tab", on page 1135.

The results describing the transient response of the system are shown in Figure 19-5 and described in Table 19-6.

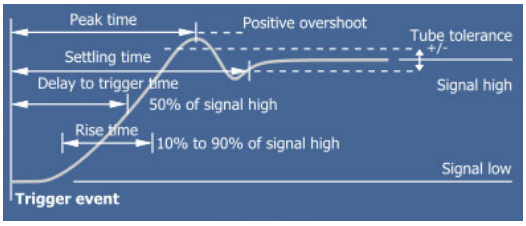


Figure 19-5: Graphical presentation of the transient response properties

Table 19-6: Transient response

Result	Description
Rise time	The time needed for the signal to change from 10% to 90% of the specified signal high.
Overshoot level	The maximum swing level above the signal high.
Settling time	The time elapsed from the trigger event to the time the output enters and remains within the tube tolerance band.
Peak time	The time needed for the response to reach the first peak of the overshoot.
Delay to trigger	The time needed for the response to reach half of the signal high value, after the trigger event.

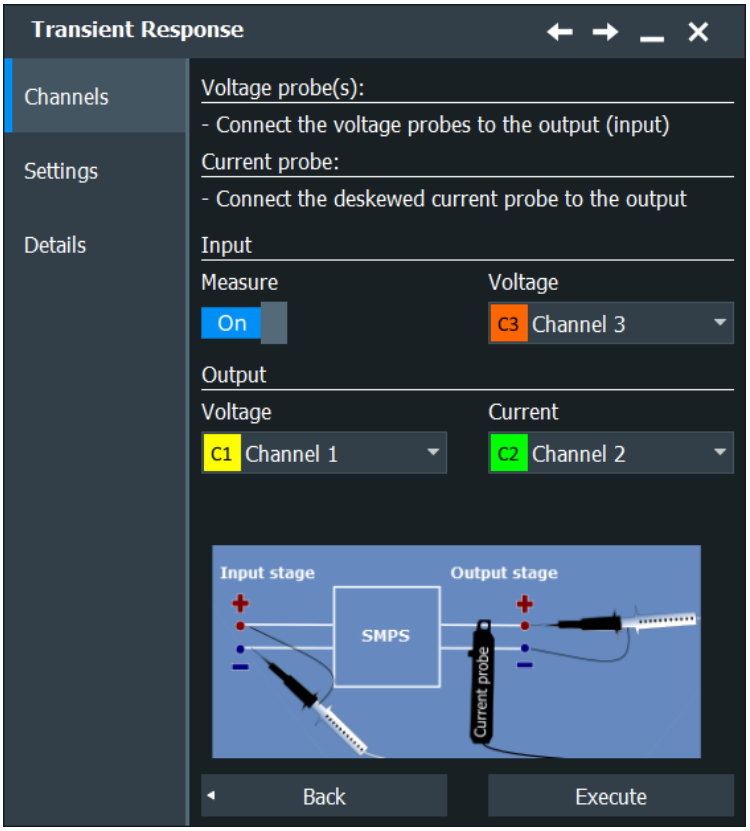
The following remote commands are used for handling the measurement results:

- `POWer:TRANsient:RESult[:ACTual]?` on page 2556
- `POWer:TRANsient:REPort:ADD` on page 2556

19.6.2.2 Configuring transient response

For details of the configuration settings, see Chapter 19.6.2.3, "Transient response settings", on page 1192.

1. Select "Analysis" > "Power".
2. Under "Output", select "Transient Response".
3. Connect one or more voltage probes and the current probe to the oscilloscope.
4. Deskew the probes as described in Chapter 19.1.1.1, "Auto deskew", on page 1123.
5. Connect the probes to the DUT as shown in the "Channels" tab:



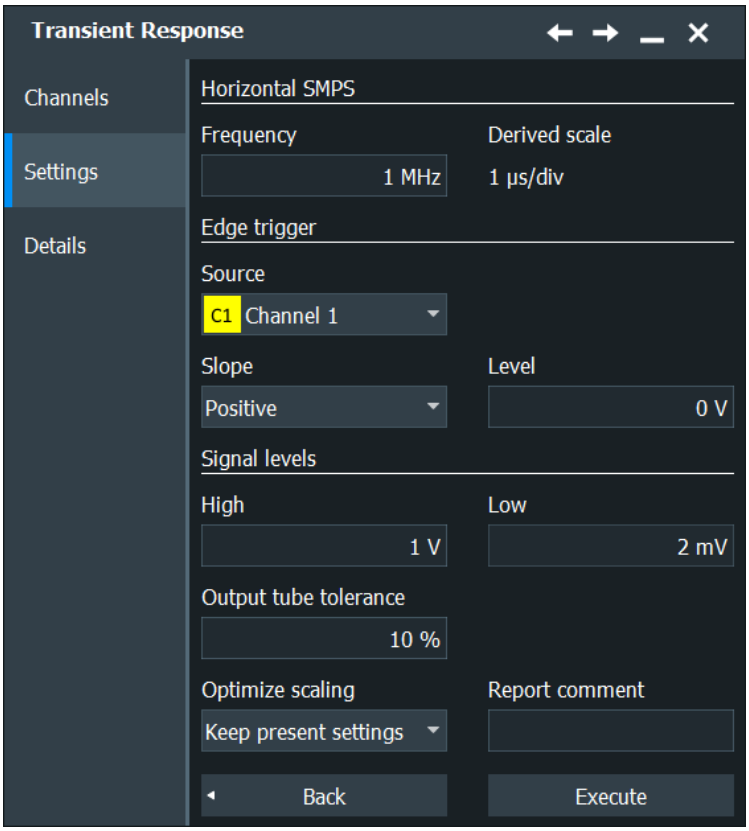
6. Select the correct channels for the "Current Source" and the "Voltage Source".
7. Select the "Settings" tab.
8. Set the "Trigger" settings according to your signal.
9. Set the "Tube tolerance", "Signal high" and "Signal low" according to your requirements.
10. Set the "SMPS switching frequency" according to your device.
11. Select an "Optimize Scaling" option.
12. Tap "Execute".
13. If needed adjust the cursors manually.

    On the screen, you can see the measurement of the current and the voltage . Also, there is a table giving information about important measurement parameters. For details, see Chapter 19.6.2.1, "Transient response results", on page 1190.

19.6.2.3 Transient response settings

In the "Channels" tab, you set the current source and the voltage sources, see also: Chapter 19.2.1, "Channels tab", on page 1134 and `POWer:TRANsient:INPut` on page 2555.

In the "Settings" tab, you configure the transient response measurement parameters and display settings.



Horizontal SMPS Frequency
Sets the SMPS switching frequency. Set the value according to your DUT.

Remote command:
`POWer:TRANsient:FREQuency` on page 2555

Derived scale
Shows the scale of the displayed results. This value is calculated as the inverse of the "SMPS switching frequency"

Edge trigger
Sets the properties of the trigger.

Source ← Edge trigger
Sets the source channel of the trigger.

Remote command:
`POWer:TRANsient:TRGChannel` on page 2556

Slope ← Edge trigger
Sets the edge type for the trigger event.

"Positive" Selects the rising edge, that is a positive voltage change.

"Negative" Selects the falling edge, that is a negative voltage change.

"Both" Selects the rising as well as the falling edge.

Remote command:
`POWer:TRANsient:TRGSlope` on page 2557

Level ← Edge trigger
Sets the voltage or current level for the trigger event.

Remote command:
`POWer:TRANsient:TRGLevel` on page 2557

Output tube tolerance
Specifies a tolerated error band for the signal level.

Remote command:
`POWer:TRANsient:HYSTeresis` on page 2555

High
Sets the expected signal high-voltage value.

Remote command:
`POWer:TRANsient:SIGHigh` on page 2556

Low
Sets the expected signal low voltage value.

Remote command:
`POWer:TRANsient:SIGLow` on page 2556

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:TRANsient:AUToscale` on page 2555

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the "Transient Response" measurement.

Remote command:
`POWer:TRANsient:EXECute` on page 2555

19.6.3 Output spectrum

"Output Spectrum" analysis measures the spectrum of the output voltage. The results can be applied to see typical side effect problems of the SMPS application, such as switching frequency components of internal SMPS.

Required probes:

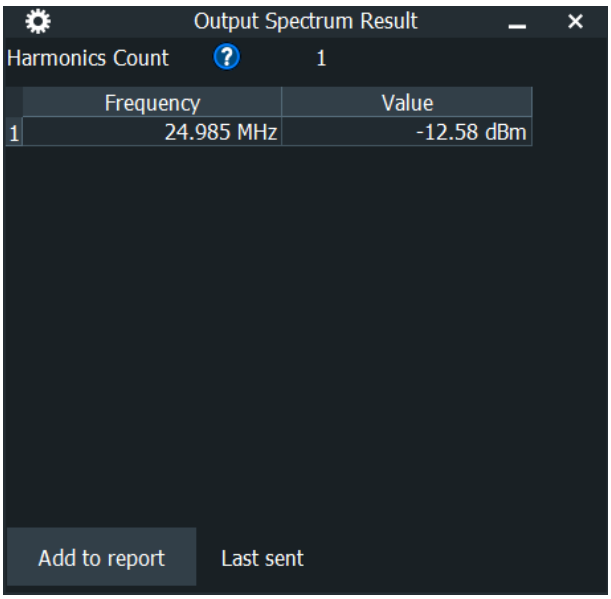
- Voltage probe

19.6.3.1 Output spectrum results

After executing the "Output Spectrum" measurement, the following windows are displayed:

The results of " Power Quality" measurements are provided in two ways:

- Two diagrams that show the graphical presentation of:
  - the voltage waveform
  - the spectrum
- The result box displays the positions of the measured peaks. The peaks are found by an automatic peak search.
  Note: If no results are found, check and correct the FFT settings.



To measure and display the output spectrum, the instrument uses the following measurements and waveforms:

- "M4" Math 4 to calculate the magnitude of the FFT for the voltage source values
- "Cu 1" to "Cu 4": Cursor 1 to Cursor 4 to determine the positions of the peaks

The used resources are listed in the "Details" tab. See also: Chapter 19.2.3, "Details tab", on page 1135.

The measured peaks have different origin. Analyzing the frequencies gives information about the influences on the output signal.

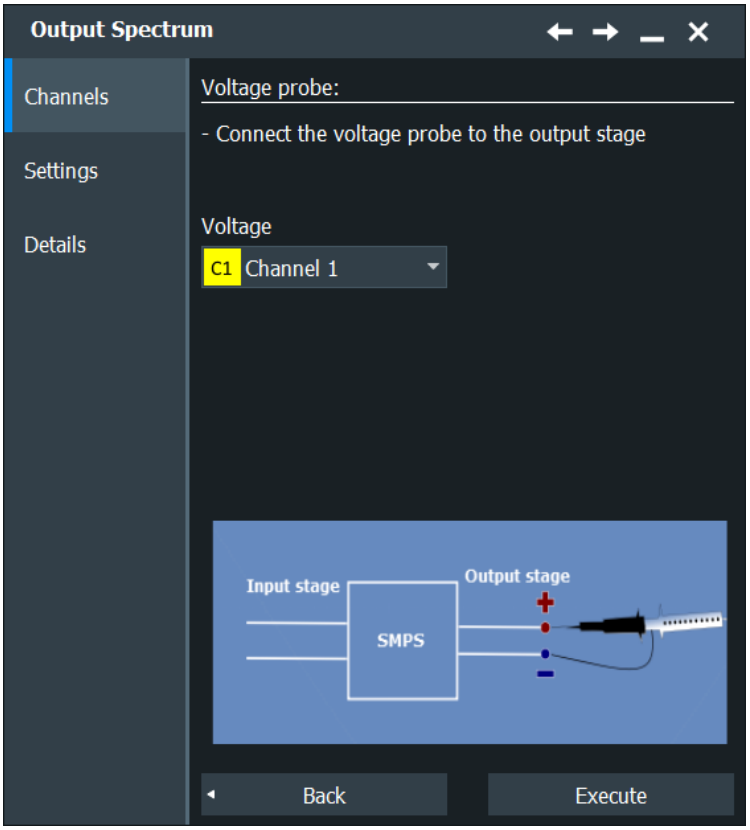
The following remote commands are used for handling the measurement results:

- `POWer:SPECtrum:RCOunt?` on page 2558
- `POWer:SPECtrum:RESult<m>:FREQuency?` on page 2558
- `POWer:SPECtrum:RESult<m>:LEVel?` on page 2558
- `POWer:SPECtrum:REPort:ADD` on page 2558

19.6.3.2 Configuring output spectrum

For details of the configuration settings, see Chapter 19.6.3.3, "Output spectrum settings", on page 1197.

1. Select "Analysis">"Power".
2. Under "Output", select "Spectrum".
3. Select the "Channels" tab.
4. Connect the probe to the DUT and to the oscilloscope as shown in the graphic:



5. Select the correct channel for the "Voltage Source".
6. Select the "Settings" tab.
7. Set the "SMPS switching frequency" according to your signal.
8. Select an "Optimize Scaling"option.

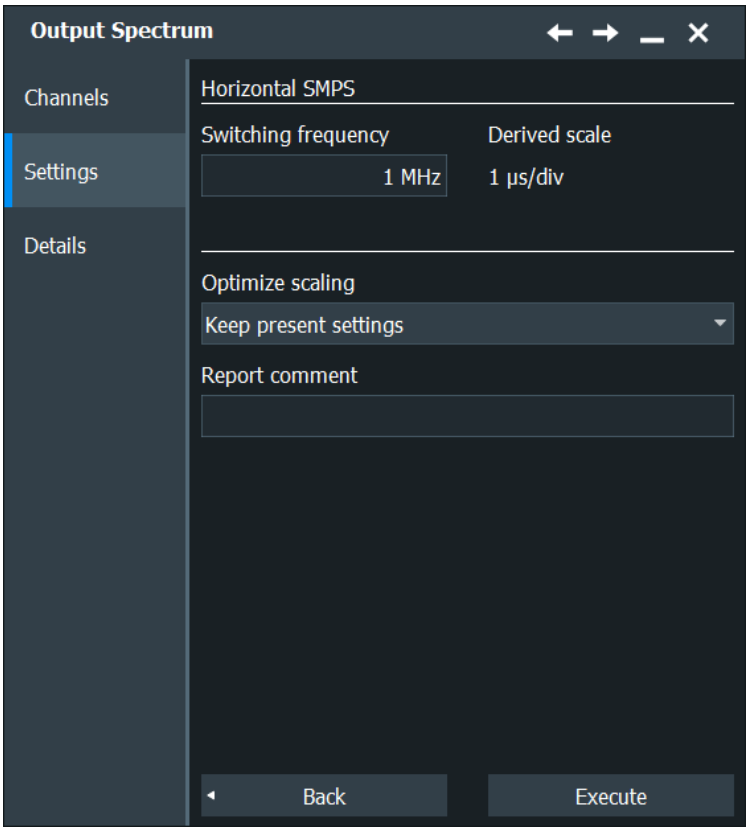
9. Tap "Execute".
10. Set the positions of the cursors according to the measured spectrum.

    On the screen, you can see the measurement of the spectrum. Also, the result box shows the position of the peaks. For details, see Chapter 19.6.3.1, "Output spectrum results", on page 1195.

19.6.3.3 Output spectrum settings

In the "Channels" tab, you set the voltage source, see also: Chapter 19.2.1, "Channels tab", on page 1134.

In the "Settings" tab, you configure the spectrum measurement parameters and display settings.



Switching frequency
Sets the SMPS switching frequency. Set the value according to your DUT.

Remote command:
`POWer:SPECtrum:FREQuency` on page 2557

Derived horizontal scale
Shows the scale of the displayed results. This value is calculated as the inverse of the "SMPS switching frequency"

Optimize scaling
Selects the scaling for the display of the results.

"Autoscale the signals"
Automatically selects the most appropriate scale for the display of the results.

"Keep present settings"
The present display settings are not changed.

Remote command:
`POWer:SPECtrum:AUToscale` on page 2557

Report comment
In this field you can write a comment that is displayed in the measurement report, for example, a measurement description.

Execute
Starts the " Output Spectrum" measurement.

Remote command:
`POWer:SPECtrum:EXECute` on page 2557

20 Deembedding (option R&S RTO6-K121)

Deembedding removes the parasitic effects of the measurement setup from the measured signal. A simple measurement setup consists of a probe only, but more complex setups include also cables, fixtures and other components. The effects of these components on the measurement are typically increasing when signal frequency increases. Thus, deembedding is useful or even necessary when measuring signals of 4 GHz frequency or higher. Furthermore, you can virtually move the measurement point to a point in a circuit that cannot be reached by probing. In this case, the effects of the components between the real and ideal probe are deembedded.

The components of a measurement setup are usually multiports, and each multiport can be described by a scattering matrix. The elements of a scattering matrix are the S-parameters. The S-parameters of a multiport are usually measured using vector network analyzers (VNA), and they are saved in Touchstone files.

The R&S RTO6 also can measure probes and cables and determine their S-parameters. These components are called proven probe or proven cable, and their characteristics are saved in `*.rs2p` files. The measurements require hardware option R&S RTO6-B7. For proven probe, also the R&S RT-ZF30 probe test fixture is needed

From the S-parameters of the measurement components, the deembedding option determines the transfer function for the measurement setup. Based on the transfer function, filter coefficients are calculated, and the filter is applied to the measured signal.

Option R&S RTO6-K121 realizes the deembedding process in software. As the process requires some time, triggering on the corrected signal is not possible, and the acquisition rate decreases.

20.1 Configuring the measurement setup for deembedding

The R&S RTO6 provides an easy way to model the measurement setup on the instrument, and to enter the characteristics of the components. If you use an R&S RT-ZM, R&S RT-ZD or R&S RT-ZS probe with Rohde & Schwarz interface, the setup is easy: The probe is recognized by the oscilloscope, and the oscilloscope knows the characteristics of the probe components.

Before you start, you need to know the following:

- Probing method: probing or terminating
- S-parameter files of the components of the measurement setup (Touchstone files). S-parameters can be listed as complex numbers (real and imaginary parts), or as magnitude and phase in the order of increasing frequency. The phase unit has to be degree (°).

Proceed as follows:

1. Open the [App Cockpit] > "Analysis" menu
2. Select "Deembedding".

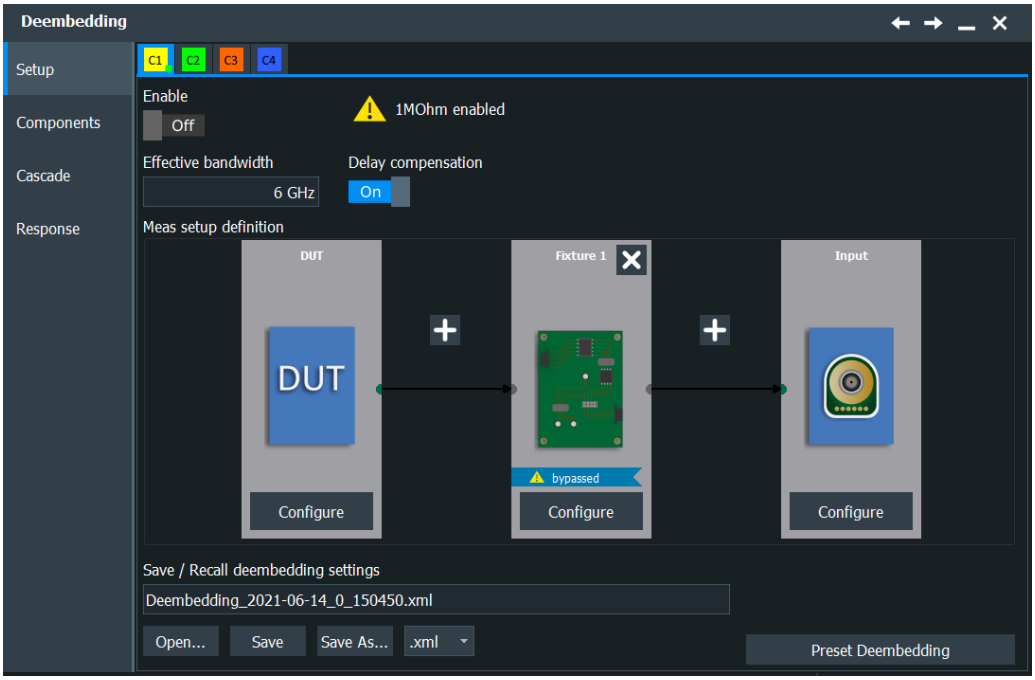
3. Select the input channel to which the measurement setup is connected.
4. Add the components of the measurement setup:
   a) Tap the "+" icon.
   b) Select the component type.

   The component is added to the setup, and it is disabled by default (bypassed). The DUT and a probe or other input component are included by default.
5. Configure each component:
   a) Tap the "Configure" button.
   b) Enter the settings as required.
      All settings are described in Chapter 20.2, "Deembedding settings", on page 1200.
   c) If necessary, tap "Open" and load the S-parameters from file.
      As long as the configuration is incomplete or conflicting, the message "Invalid configuration" is shown.
   d) Enable the component to consider it during deembedding.
   e) Tap the subtab of the next component, and configure it in the same way.
6. On the "Setup" tab, set the Effective bandwidth and the Delay compensation.
7. Enable the deembedding.
8. If you want to keep the configuration for later use, save it to file.

20.2 Deembedding settings

20.2.1 Measurement setup

In the "Deembedding" > "Setup" tab, you add all components of the measurement setup and define all settings that are related to the complete path.



Enable
Activates the deembedding - the correction of parasitic effects of the measurement setup on the measured signal.

Remote command:
`DEEMbedding<m>[:STATe]` on page 2559

Effective bandwidth
Sets the maximum bandwidth until which the signal is corrected. This maximum value is the minimum bandwidth value of the probe bandwidth, tip module bandwidth (R&S RT-ZM) and oscilloscope bandwidth, and can not be higher than the highest frequency in a used S-parameter file. The setting is also coupled with the digital filter bandwidth if the cut-off frequency of the digital filter is at least 1 GHz.

Remote command:
`DEEMbedding<m>:BANDwidth` on page 2560

Delay compensation
Enables the compensation for the group delay of the complete measurement setup. If enabled, the calibration time reference point is the start point of the measurement setup. If disabled, the calibration time reference point is the oscilloscope's channel input.

Remote command:
`DEEMbedding<m>:CMPDelay` on page 2561

Meas setup definition
Simulates the complete measurement setup with all components.

To add a component, tap the **+** icon and select the component type: cable, fixture, adapter, proven probe, proven cable, or others ("Custom"). The DUT component is always included in the deembedding. Also, a connected R&S RT-ZM, R&S RT-ZD or R&S RT-ZS probe with Rohde & Schwarz interface is automatically recognized and included in the deembedding.

The remark "bypassed" indicates that the component is disabled and not considered in the deembedding calculation as long as it is not configured correctly.

Remote command:
`DEEMbedding<m>:ADD` on page 2559
`DEEMbedding<m>:REMove` on page 2560
`DEEMbedding<m>:CCOunt?` on page 2560

Save / recall deembedding settings
Saves the complete deembedding configuration, or loads a saved deembedding configuration. The file format is XML.

To save the deembedding configuration, enter a filename and tap "Save". Alternatively, tap "Save as" to open the file selection dialog box, see Chapter 12.5, "File selection dialog", on page 482.

To load a deembedding configuration, tap "Open" and select the required file.

Remote command:
`DEEMbedding<m>:NAME` on page 2561
`DEEMbedding<m>:SAVE` on page 2562
`DEEMbedding<m>:OPEN` on page 2562
`DEEMbedding<m>:DELete` on page 2562

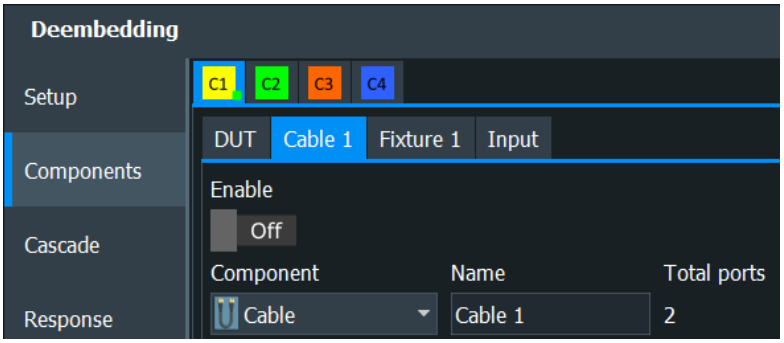
Preset Deembedding
Sets the complete deembedding configuration to the default values.

Remote command:
`DEEMbedding<m>:PRESet` on page 2561

20.2.2 Common settings for components

The "Components" tab has several subtabs to configure each component of the measurement setup that was defined on the "Setup" tab.

Settings that are available on all subtabs are described in this chapter. Other settings are component-specific and explained in the following chapters.



Enable
Enables the component for deembedding. As long as the configuration is incomplete or conflicting, the message "Invalid configuration" is shown, and enabling is not possible.

Remote command:
`DEEMbedding<m>:COMPonent<n>[:STATe]` on page 2562

Component
Selects the component type. The "DUT", "Input" and "Probe" types are predefined and shown for information. The other types ("Fixture", "Cable", "Adapter", and "Other") are selectable.

Remote command:
`DEEMbedding<m>:COMPonent<n>:TYPE` on page 2563

Name
Enter a name for the component that helps you identifying it.

Remote command:
`DEEMbedding<m>:COMPonent<n>:NAME` on page 2563

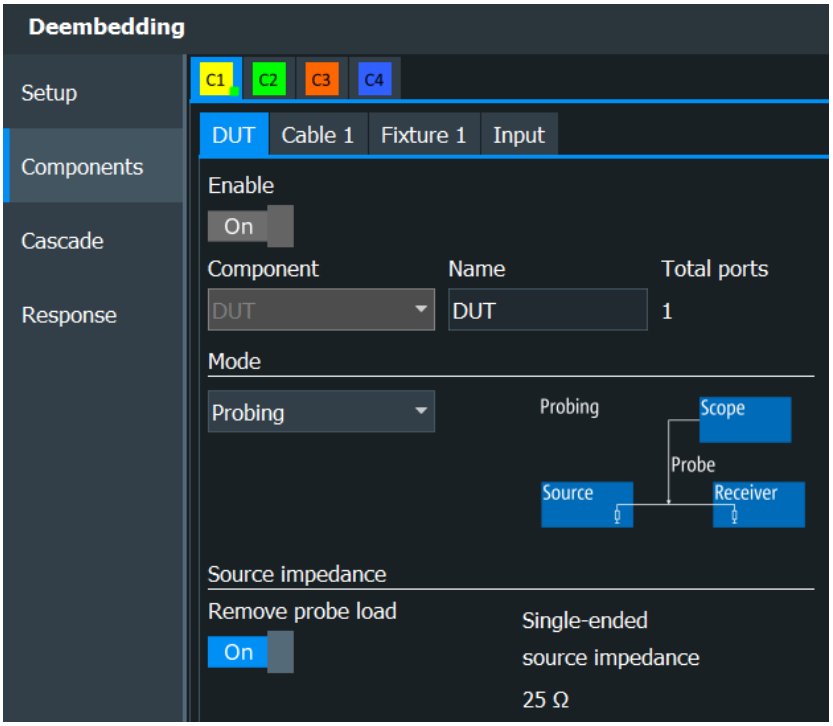
Total ports
Shows the number of all ports of the selected component (input ports and output ports).

The number of ports depends on the attached probe. If a modular probe R&S RT-ZM or differential probe R&S RT-ZD is used, the number of ports is 4. Otherwise, 2 ports are available.

Remote command:
`DEEMbedding<m>:COMPonent<n>:PCOunt?` on page 2564

20.2.3 DUT settings

In the "Deembedding" > "Components" > "DUT" tab, you set the characteristics of the DUT.



Probing, Terminating
Sets the probing mode.

Select "Probing" if you use a high-ohmic probe and measure on an existing line in parallel to the load.

Select "Terminating" if the measuring equipment is the load of the line.

Remote command:
`PROBe<m>:DEEMbedding:MODE` on page 1360
`DEEMbedding<m>:COMPonent<n>:MODE` on page 2564

Remove probe load
If "Probing" is selected, you can remove the loading of the probe.

If the probe loading is removed, you see the signal that would be at the measurement point if the probe's input impedance is ideal (infinite impedance).

Without removing the loading, you see the real signal at the measurement point, including the probe loading.

Remote command:
`PROBe<m>:DEEMbedding:REMProbeload` on page 1360
`DEEMbedding<m>:COMPonent<n>:LOAD:REMProbeload` on page 2565

Source impedance
Shows the source impedance of the DUT. If an R&S RT-ZM probe is connected, the value depends on the selected probe mode: common mode, differential, or single-ended measurement.

Remote command:
`DEEMbedding<m>:COMPonent<n>:LOAD:IMPedance?` on page 2565
`PROBe<m>:DEEMbedding:LOAD:IMPedance` on page 1361

20.2.4 R&S RT-ZM probe settings

If an R&S RT-ZM modular probe is connected to a channel input, the probe is detected by the oscilloscope and automatically added to the measurement path. The S-parameters of the probe, the tip modules, and the probe group delay are known to the oscilloscope and used for deembedding.

Probe settings that are important for deembedding can be set directly in the deembedding dialog.

To get all probe deembedding settings:

1. In the "Deembedding" > "Components" > "RT-ZMxx" tab, select "More Settings".
2. In the "Probes Setup" dialog box, select "Probe Deembedding".

The "Probe Deembedding" dialog box is described in Chapter 5.6.5.2, "Deembedding for modular probes", on page 159.

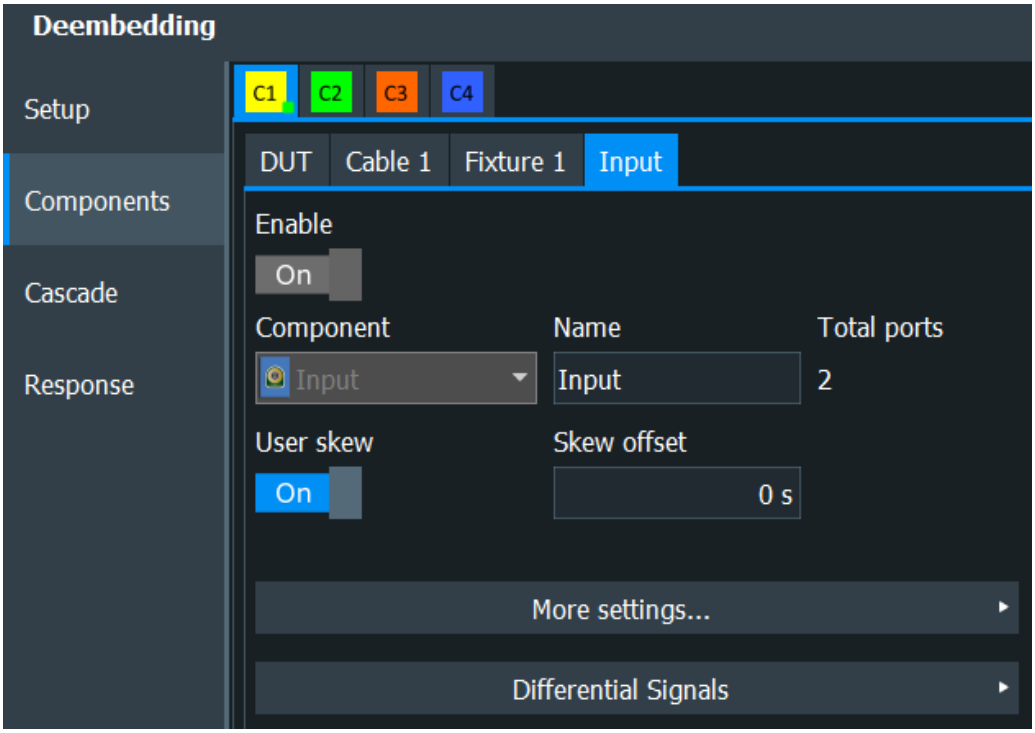
For details on probe parameters, see:

- "Select probe tip module, Pin spacing" on page 159
- "Probe mode" on page 158
- "DM Offset, CM Offset, P Offset, N Offset" on page 159

20.2.5 Input settings for other probes

The "Input" component is used for all probes except for R&S RT-ZM, R&S RT-ZD and R&S RT-ZS probes with Rohde & Schwarz interface, and for cables or other components that are used to connect the measurement setup to the oscilloscope. For these input components, the horizontal skew and vertical settings are considered during deembedding. You can set the skew directly in the "Deembedding" > "Components" > "Input" tab, or in "Horizontal" menu > "Skew". The button "More Settings" opens the "Channels" settings.

The input component is also used for deembedded differential signals, see Chapter 5.8.3, "Deembedding of differential signals", on page 184.



For details, see:

- "User skew" on page 124
- "Skew offset" on page 124

20.2.6 Cable, adapter, fixture and custom settings

If you have added a cable, adapter, test fixture or other component to the measurement setup, you set the characteristics in the corresponding subtabs of the "Deembedding" > "Components" tab.

The characteristics are the same for these component types. Configuration differs between 2-port and 4-port components, because 4-ports need additional settings. Generally, these components are configured by loading S-parameter files and analyzing its characteristics.

To configure a component in a single-ended 2-port setup

1. Tap "Open".
2. Select and load the `*.s2p` Touchstone file that contains the S-parameters of the component. The filetype is always `*.s2p`.
3. Assign the ports to the input and output of the component according to the data in the Touchstone file. See also: "Input 1, Input 2, Output 1, Output 2" on page 1208.
4. Analyze the response curves of each S-parameter. The response curves are functions of magnitude, phase, and group delay over frequency.
   a) Select the S-parameter.

b) Check the response curves. The display is based on a reference impedance of Z0 = 50 Ω.

The response curves show:
- The correctness of the S-parameter file.
- The correct setting of the input and output ports ("Input 1", "Output 1")
- The characteristics of the component.

5. If the component is configured correctly, tap "Enable" to include it in the deembedding setup.

To configure a component in a single-ended 4-port setup

1. If the component is a 4-port, i.e. a fixture or custom block, load the `*.s4p` Touchstone file that contains the S-parameters of the component:
   a) Tap "Open".
   b) Select and load the `*.s4p` Touchstone file. The filetype is always `*.s4p`.
2. If the component is a 2-port, i.e. a cable or adapter, typically 2 cables or adapters are used. These components can be different or identical.
   a) If the components are identical, enable "Identical Paths" on page 1208. You need 1 `*.s2p` Touchstone file.
   If the components are different, disable "Identical Paths" on page 1208. You need 2 `*.s2p` Touchstone files, one for each component.
   b) Select the filetype `*.s2p`.
   c) Tap "Open".
   d) Select and load the `*.s2p` Touchstone file.
   e) If the components are different, load the second `*.s2p` Touchstone file in the same way.
3. Assign the ports to the input and output of the component according to the data in the Touchstone file. See also: "Input 1, Input 2, Output 1, Output 2" on page 1208.
4. Analyze the response curves of each S-parameter. You can analyze magnitude, phase and group delay response, which are functions of frequency.
   a) Select the S-parameter.
   b) Check the response curves. The display is based on a reference impedance of Z0 = 50 Ω.

   The response curves show:
   - The correctness of the S-parameter file.
   - The correct setting of the input and output ports ("Input 1", "Output 1")
   - The characteristics of the component.
5. If the component is configured correctly, tap "Enable" to include it in the deembedding setup.

Response curves are described in Chapter 20.3.1, "Responses of the component cascade", on page 1209.

Input 1, Input 2, Output 1, Output 2

Assign the ports to the input and output of the component according to the data in the Touchstone file. This information is not directly written in the file. You need background information how the S-parameters were determined, i.e. which port was input and which port was output. If in doubt, the response curves help assigning the ports correctly.

Remote command:

`DEEMbedding<m>:COMPonent<n>:IPONe` on page 2567
`DEEMbedding<m>:COMPonent<n>:IPTWo` on page 2567
`DEEMbedding<m>:COMPonent<n>:OPONe` on page 2567
`DEEMbedding<m>:COMPonent<n>:OPTWo` on page 2567

Identical Paths

The setting is relevant if the measurement setup is a 4-port, and the component is a 2-port. Usually, these components are cables and adapters, which are described by `*.s2p` files. If there is a 4-port setup, 2 cables or 2 adapters are used. If there is no crosstalk in-between, each cable or adapter can be described by a `*.s2p` file.

If both components are different, each is described by its own `*.s2p` file. If they are identical, for example, when a matched-pair cable is used, they can be described by the same file. In this case, enable "Identical Paths".

Remote command:

`DEEMbedding<m>:COMPonent<n>:IDENtical` on page 2567

S-Parameters, Filetype

Loads the Touchstone file that contains the S-parameters of the component.

If the measurement setup is a 2-port, the filetype is always `*.s2p`.

If the measurement setup is a 4-port, the selection depends on the component. Fixtures and custom blocks are usually 4-ports, and they are described by `*.s4p` S-parameter files.

Cables and adapters are usually 2-ports, and they are described by `*.s2p` S-parameter files. If there is a 4-port setup, 2 cables or 2 adapters are used. You need 2 different files if the components are different, or only 1 file if they are identical (see also "Identical Paths" on page 1208).

Remote command:

`DEEMbedding<m>:COMPonent<n>:SPONe` on page 2568
`DEEMbedding<m>:COMPonent<n>:SPTWo` on page 2568

20.3 Responses

20.3.1 Responses of the component cascade

The "Deembedding" > "Cascade" tab combines the S-parameters of all included components and shows the response curves of the measurement setup for each S-parameter:

- Magnitude response
- Phase response
- Group delay response

While the response curves on the component tabs show the S-parameter characteristics of a single component, the cascade shows the combined characteristics for all components including the mismatch between the components. The DUT is not considered.

Port selection
Selects the S-parameter. The response curves (magnitude, phase, and group delay response) of the setup cascade for the selected parameter are shown. You can refresh each graphic, and display it on full screen.

20.3.2 Responses of the measurement setup

On the "Deembedding" > "Response" tab, you see the resulting transfer function characteristics of the deembedded measurement setup. These response curves include also the characteristics of the DUT and its interaction with the input connector of the oscilloscope.

You can see the following characteristics:

- Magnitude response
- Phase response
- Group delay response
- Step response

Using these response curves, you can assess the correction of the signal by deembedding. In particular, you can compare realtime deembedding with software deembedding and check whether the accuracy of the realtime deembedding is sufficient for the measurement task.

► To see more details in a response graphic, tap the upper right button in the graphic.

The graphic opens in full view, where you can analyze the response function: zoom in and out, use the cursor.

On the "Response" tab, you see also some parameters of the uncorrected measurement setup including the DUT. All parameters are related to a frequency range from DC to the configured effective bandwidth.

Att_{max}
Maximum attenuation of the measurement setup before deembedding, with reference to a frequency response normalized at DC to 0 dB. The deembedding filter must amplify the amplitude response by the "Att_{max}" value, and thus also increases the noise by this value. The value also helps to decide whether deembedding is useful for the defined effective bandwidth.

Remote command:
`DEEMbedding<m>:MATTenuation?` on page 2569

Att_{DC}, $Gain_{DC}$
Maximum attenuation or gain of the measurement setup before deembedding at DC. This value is the expected basic attenuation or gain, which is corrected by deembedding.

Remote command:
`DEEMbedding<m>:CADC?` on page 2568

$Gain_{max}$
Maximum gain of the measurement setup before deembedding, with reference to a frequency response normalized at DC to 0 dB. The deembedding filter must attenuate the amplitude response by this value. The value helps also to avoid an overload of the ADC.

Remote command:
`DEEMbedding<m>:MGAin?` on page 2569

20.4 Proven probe, proven cable

The R&S RTO6 can measure probes and cables and determine their S-parameters. These components are called proven probe or proven cable, and their characteristics are saved in `*.rs2p` files. The measurements require hardware option R&S RTO6-B7. For proven probe, also the R&S RT-ZF30 probe test fixture is needed

20.4.1 Characterization of proven probes and cables

To get the S-parameters of proven probes and cable, several measurements are needed. A wizard guides you through the process, and graphics show how to connect the setup. You can also save the result of each step and speed up further measurements by saved step results.

The wizard steps are the following:

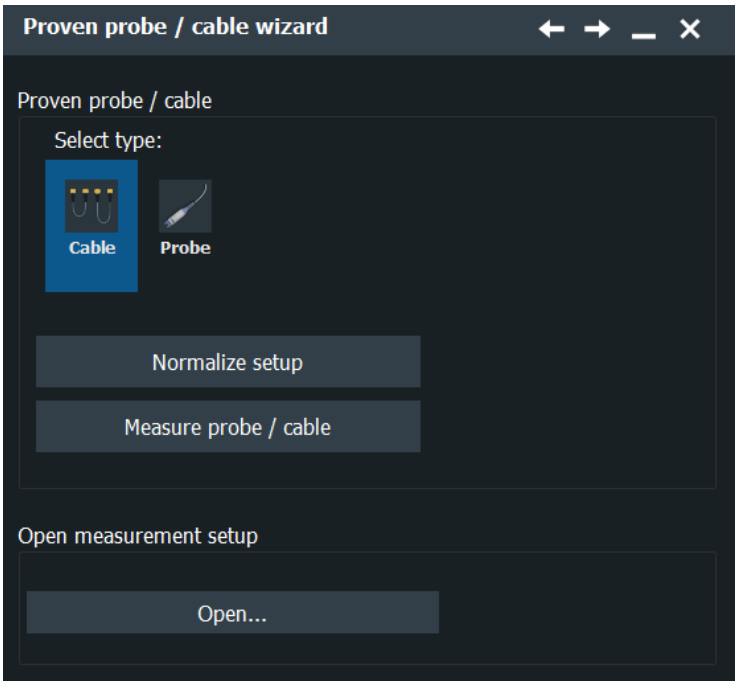
- Proven probe / cable setup 1211
- Normalization 1212
- Measurement 1213
- Response 1214

20.4.1.1 Proven probe / cable setup

Access: [App Cockpit] > "Analysis" > "Deembedding" > "Setup" tab: include component "Proven Probe" or "Proven Cable" > "Configure" > "Start wizard"

Depending on the steps that you have done before, you can:

- Start the complete characterization procedure.
- Start the probe or cable measurement in a normalized setup.
- Load measurement results.



Select type
Select the DUT to be characterized: probe or cable

Normalize setup
Startes the wizard for the complete normalization and measurement procedure.

Measure probe / cable
Starts the measurement of the probe or cable directly. If the measurement setup has been normalized just before, the "Measurement" dialog box opens. Otherwize, you are asked to open a normalization file, or to start with normalization.

Open measurement setup
Opens a file with previously saved measurement results. The "Response" dialog box opens, and you can check the response plots, and save or apply the S-parameters to the deembedding setup.

20.4.1.2 Normalization

Open [App Cockpit]> "Analysis" tab > "Proven Probe / Cable" > "Normalize setup"

1. Connect the cables to the pulse source R&S RTO6-B7 and to the input channels.
   The probe or cable to be measured is *not* connected.
   Use only high-quality double-shielded RF cables. Two suitable cables are delivered with the pulse source option R&S RTO6-B7.

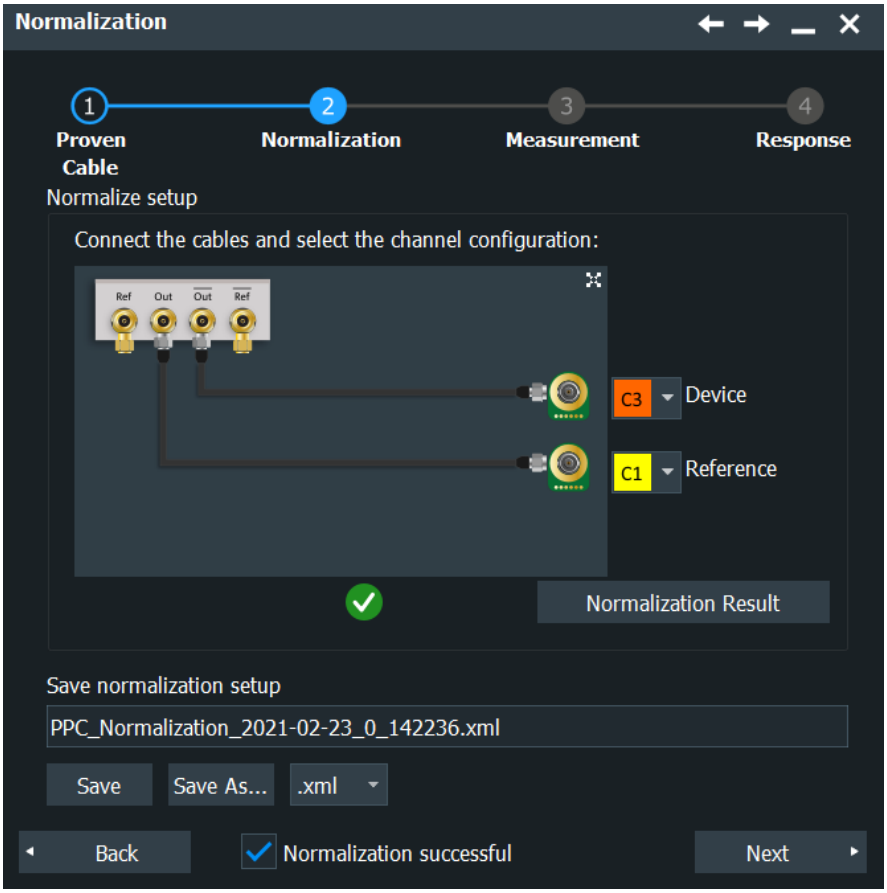


Figure 20-1: Normalization settings

2. Start measurements with "Normalize".

   The basic setup is characterized.

Device
Selects the input channel that is connected to the $\overline{\text{Out}}$ connector of the pulse source R&S RTO6-B7 during normalization, and to the measured probe or cable during measurements.

Reference
Selects the input channel that is connected to Out connector of the pulse source.

Normalize
Starts the normalization of the setup without the probe or cable to be measured. The frequency correction and deskew are determined and applied in the next measurement step.

If normalization was successful, it is indicated by an icon, and you can display the "Normalization Result" - the normalized magnitude response.

Save normalization setup
Saves the results of the normalization step to an XML file.

20.4.1.3 Measurement

Open [App Cockpit]> "Analysis" tab > "Proven Probe / Cable" > "Normalize setup"> "Next"

1. To characterize a cable, connect it to the cable coming from the $\overline{\text{Out}}$ connector of the pulse source and the "Device" input channel. Use an adapter if necessary.

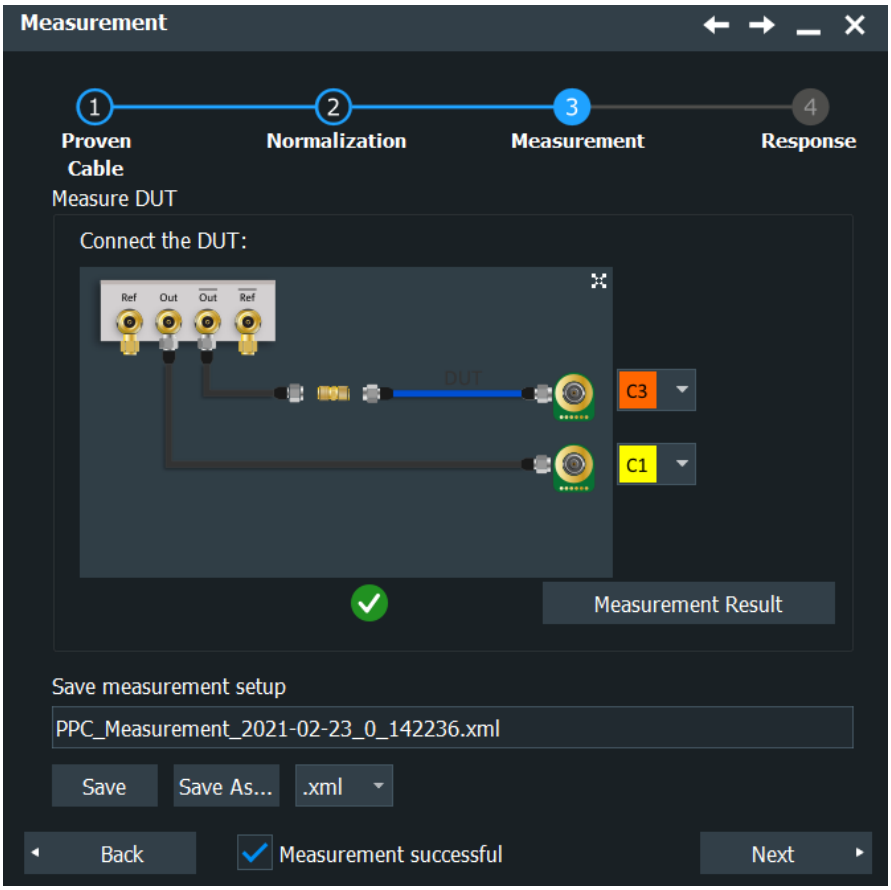


Figure 20-2: Cable measurement

2. To characterize a probe, you need the R&S RT-ZF30 probe test fixture.

a) Connect the fixture to the cable coming from the $\overline{\text{Out}}$ connector of the pulse source.
b) Connect the probe to the fixture.
For details, see the "R&S RT-ZF30 Probe Test Fixture Manual".
c) Connect the probe to the "Device" input channel.

3. Select "Measure".

Measure
Starts the measurement of the probe or cable to be characterized.

If the measurement was successful, it is indicated by an icon, and you can display the "Measurement Result" - the magnitude response of the proved probe or cable.

Save measurement setup
Saves the measurement results of the proven probe or cable to an XML file.

20.4.1.4 Response

Open [App Cockpit]> "Analysis" tab > "Proven Probe / Cable" > > "Normalize setup" > "Next" > "Measure" > "Next"

In the last step, you see the response curves, the charcteristics of deembebbed proven probe or cable. The characteristics are the same as on the "Deembedding Response" tab. For more information, see Chapter 20.3.2, "Responses of the measurement setup", on page 1209.

Apply result
Includes the proven probe or proven cable component in the deembedding setup, and opens the "Deemebedding Components" tab for configuration.

Save result
Saves the S-parameters of the proven probe or cable to a rs2p or rs4p file. Using this file, you can include and configure this proven probe or cable as a component in a deembedded measurement setup.

20.4.2 Deembedding with proven probe or proven cable

If a probe or cable is characterized as described in Chapter 20.4.1, "Characterization of proven probes and cables", on page 1210, and its S-parameters are saved, you can use this probe or cable in any deembedding measurement setup.

1. In the deembedding measurements setup, add the component "Proven Probe" or "Proven Cable".
2. Select "Configure" on the new component.
3. Load the S-parameters file (`*.rs2p`).
If no file is available, you can start the characterization wizard here.

See also: Chapter 20.1, "Configuring the measurement setup for deembedding", on page 1199.

21 Compliance tests

Thanks to the high measurement accuracy of the R&S RTO6, the oscilloscope can be used for reliable compliance testing on the physical layer of high-speed bus interfaces. For fast and easy compliance testing, you need the free-of-charge R&S ScopeSuite. The software guides you step-by-step through the test sequences, configures the oscilloscope, automatically performs measurements and compiles the results in a measurement report. Detailed, image-based instructions make it easy to connect the oscilloscope and the probes to the test fixture set and the device under test.

The R&S ScopeSuite can be installed on a test computer or directly on the R&S RTO6. If the R&S ScopeSuite is installed on the oscilloscope, you can start it directly in the firmware menu.

For best operation results, we recommend that the installed firmware versions of the R&S ScopeSuite and the oscilloscope are the same.

Access: [App Cockpit] > "Compliance" tab

21.1 USB compliance test

The R&S RTO6-K21 option enables the automatic testing of USB 2.0 compliance (high-speed) as well as USB 1.1 (full-speed) and USB 1.0 (low-speed).

As an example, the test setup for USB device tests is shown below.

The test fixture and the USB compliance test procedures are described in separate manuals:

- "R&S®RTO-K21, R&S®RTP-K21/-101 R&S®USB 2.0/3.2 Compliance Test, User Manual"
- "R&S®RT-ZF1 USB 2.0 Compliance Test Fixture Set, Manual"

21.1.1 Test equipment

USB 2.0 test equipment

For USB 2.0 compliance tests, the following test equipment is recommended:

- R&S RTO6 oscilloscope with 4 channels and at least 2 GHz bandwidth.
  For full speed and low speed compliance tests, a 4-channel R&S RTO with less than 2 GHz bandwidth is sufficient.
- R&S RT-ZF1 USB 2.0 compliance test fixture set, which consists of the load board and the signal quality board.
- Probes:
  – 2x single-ended active probes with at least 2 GHz bandwidth
  – 1x differential active probe with at least 2 GHz bandwidth
- R&S RTO-B6/R&S RTP-B6/Tabor WX2182B/Tabor WX2182C arbitrary waveform generator for automatic receiver sensitivity tests.
  For manual receiver sensitivity tests, any AWG can be used.
- R&S ScopeSuite software, which can be installed on a computer or directly on the R&S RTO6.
- R&S RTO6-K21 USB 2.0 compliance test option (required option, installed on the oscilloscope)
- Make sure that you use the latest version of the USBHSET tool. To find downloads, search for *USBHSET* on the following websites:
  – USB 3.0 host: www.usb.org/developers/tools/
  – USB 2.0 host: www.usb.org/developers/tools/usb20_tools/

  Regularly check the USB-IF website compliance.usb.org/index.asp?UpdateFile=Electrical&Format=Standard for USB-IF compliance updates.

21.2 Ethernet compliance tests (options R&S RTO6-K22/K23/K24/K87/K88/K89)

The Ethernet compliance test options enable the automatic testing of various Ethernet standards:

- 10/100/1000BASE-T (R&S RTO6-K22)
- 2.5/5/10GBASE-T (R&S RTO6-K23)

- 100BASE-T1: (R&S RTO6-K24)
- 1000BASE-T1: (R&S RTO6-K87)
- 1000BASE-T1: (R&S RTO6-K88
- 10BASE-T1: (R&S RTO6-K89)

The test fixture and the Ethernet compliance test procedures are described in the separate manuals:

- R&S®RTO, R&S®RTP Ethernet Compliance Test, User Manual"
- "R&S®RT-ZF2 Ethernet Compliance Test Fixture Set Manual"
- "R&S®RT-ZF3 Frequency Converter Manual
- "R&S®RT-ZF4 10BASE-Te Fixture Manual
- R&S®RT-ZF5 Ethernet Probing Fixture Manual
- R&S®RT-ZF6 Frequency Converter Manual

21.2.1 Test equipment

For Ethernet compliance tests, the following test equipment is needed:

- The free-of-charge R&S ScopeSuite software, which can be installed on a computer or directly on the oscilloscope.
- **10/100/1000BASE-T Ethernet compliance tests:**
  - R&S RTO/R&S RTP oscilloscope with at least 600 MHz bandwidth
  - Differential probe with at least 1 GHz bandwidth
  - R&S RTO/R&S RTP-K22 10/100/1000BASE-T Ethernet compliance test option (required option, installed on the oscilloscope)
  - R&S RT-ZF2 Ethernet test fixture set
  - For energy-efficient Ethernet tests, in addition:
    R&S RT-ZF5 Ethernet test fixture set for 100/1000BASE-T EEE tests
    R&S RT-ZF4 test fixture for 10BASE-Te tests
- **2.5GBASE-T Ethernet compliance tests:**
  - R&S RTO/R&S RTP oscilloscope with at least 600 MHz bandwidth
    For precise power spectral density and power level measurements up to 1 GHz, it is recommended to use an oscilloscope and a differential probe with 1 GHz bandwidth each.
  - Differential probe with at least 1GHz bandwidth
  - R&S RTO/R&S RTP-K23 GBASE-T Ethernet compliance test option (required option, installed on the R&S RTO6)
  - R&S RT-ZF2 Ethernet test fixture set
- **5GBASE-T Ethernet compliance tests:**
  - R&S RTO/R&S RTP oscilloscope with at least 1 GHz bandwidth
    For precise power spectral density and power level measurements up to 1.5 GHz, it is recommended to use an oscilloscope and a differential probe with 1.5 GHz bandwidth each.
  - Differential probe with at least 1.5 GHz bandwidth

  - R&S RTO/R&S RTP-K23 GBASE-T Ethernet compliance test option (required option, installed on the R&S RTO6)
  - R&S RT-ZF2 Ethernet test fixture set
- **10GBASE-T Ethernet compliance tests:**
  - R&S RTO/R&S RTP oscilloscope with at least 2 GHz bandwidth
    For precise power spectral density and power level measurements up to 3 GHz, it is recommended to use an oscilloscope and a differential probe with 3 GHz bandwidth each.
  - Differential probe with at least 3 GHz bandwidth
  - R&S RTO/R&S RTP-K23 GBASE-T Ethernet compliance test option (required option, installed on the R&S RTO6)
  - R&S RT-ZF2 Ethernet test fixture set
- **100BASE-T1 compliance tests:**
  - R&S RTO/R&S RTP oscilloscope with at least 600 MHz bandwidth
  - Differential probe with 1 GHz bandwidth
  - R&S RTO/R&S RTP-K24 100BASE-T1 compliance test option (required option, installed on the R&S RTO6)
  - R&S RT-ZF2 Ethernet test fixture set
  - R&S RT-ZF3 frequency converter for transmitter distortion tests
- **1000BASE-T1 compliance tests:**
  - R&S RTO/R&S RTP oscilloscope with at least 2 GHz bandwidth
  - Differential probe with 2 GHz bandwidth
  - R&S RTO/R&S RTP-K87 1000BASE-T1 compliance test option (required option, installed on the R&S RTO6)
  - R&S RT-ZF2 Ethernet test fixture set
  - R&S RT-ZF6 frequency converter for transmitter distortion tests
- **MGBASE-T1 compliance tests:**
  - R&S RTO / R&S RTO6 /R&S RTP oscilloscope, with a bandwidth of:
    - for 2.5 GBASE-T1 tests: at least 3.5 GHz
    - for 5 GBASE-T1 tests: at least 7 GHz
    - for 10 GBASE-T1 tests: at least 14 GHz
  - 2 BNC/SMA cables
  - R&S RTO/R&S RTP-K88 MGBASE-T1 compliance test option (required option, installed on the R&S RTO6)
- **10BASE-T1 compliance tests:**
  - R&S RTO/R&S RTP oscilloscope
  - Optional Differential probe with 1 GHz bandwidth
  - R&S RTO/R&S RTP-K89 10BASE-T1 compliance test option (required option, installed on the R&S RTO6)
- R&S RTO-B6/R&S RTP-B6/Tabor WX2182B/Tabor WX2182C arbitrary waveform generator for automatic disturber tests. For some disturber tests, also HMF2550 can be used.

- R&S ZNB/ZNC/ZND/ZVL vector network analyzer for automatic return loss measurements. For manual measurements, also other AWGs and VNAs can be used.

21.3 D-PHY compliance tests (option R&S RTO6-K26)

The option R&S RTO6-K26 enables the automatic processing of D-PHY compliance tests.

The D-PHY compliance test procedures are described in a separate manual:

- R&S®RTO-K26, R&S®RTP-K26, D-PHY Compliance Test, User Manual"

21.3.1 Test equipment

For D-PHY compliance tests, the following test equipment is needed:

- R&S RTO6 oscilloscope with 4 channels and at least 4 GHz bandwidth
- For measuring the clock signal (+ and -): either 1 differential probe or 2 single-ended probes with at least 4 GHz bandwidth. However, note that D-PHY Group 2 and Group 4 tests require 2 probes for the clock signal.
- For measuring the data signal (+ and -): 2 probes with at least 4 GHz bandwidth
- R&S RTO6-K26 D-PHY compliance test option (required option, installed on the oscilloscope)
- R&S RTO6-K27 D-PHY compliance test option (required option, installed on the oscilloscope)
- For Eye Tests, the following options are needed:

  For R&S RTO/ R&S RTO6:

  – R&S RTO/R&S RTO6-K91 (DDR3) or R&S RTO/R&S RTO6-K93 (DDR4)
- Recommended test fixture for LP-TX tests: MIPI D-PHY Capacitive Load (C_{LOAD}) fixture from The University of New Hampshire InterOperability Laboratory (UNH-IOL)
- Recommended termination board for HS-TX tests: MIPI D-PHY Reference Termination Board (RTB) from The University of New Hampshire InterOperability Laboratory (UNH-IOL)
- The free-of-charge R&S ScopeSuite software, which can be installed on a computer or directly on the oscilloscope.

21.4 eMMC compliance tests (option R&S RTO6-K92)

The option R&S RTO6-K92 enables the automatic processing of eMMC compliance tests.

The eMMC compliance test procedures are described in a separate manual:

- eMMC Compliance Tests, Test Procedures Manual"

21.4.1 Test equipment

For eMMC compliance tests, the following test equipment is needed:

- R&S RTO6 oscilloscope with at least 1 GHz bandwidth
- A probe with at least 1.5 GHz bandwidth
- R&S RTO6-K92 eMMC compliance test option (required option, installed on the R&S RTO6)
- The free-of-charge R&S ScopeSuite software, which can be installed on a computer or directly on the R&S RTO6.

21.5 PCIe compliance tests (option R&S RTO6-K81)

The option R&S RTO6-K81 enables the automatic processing of PCIe 1.1 / PCIe 2.0 compliance tests.

The PCIe compliance test procedures are described in a separate manual:

- R&S®RTO-K81, R&S®RTP-K81/-K83, PCIe Compliance Test, User Manual"

21.5.1 Test equipment

PCIe 1.1 / PCIe 2.0 Test Equipment

For PCIe compliance tests, the following test equipment is needed:

- R&S RTO6 oscilloscope with at least 6 GHz bandwidth, 2 channels and 20 GS/s available
- Probes/cables:
  - Two SMP to SMA cables or
  - Two single-ended probes with at least 6GHz bandwidth bandwidth or
  - One differential probe with at least 6GHz bandwidth bandwidth
- R&S RTO6-K81 PCIe compliance test option (required option, installed on the oscilloscope)
- Recommended test fixture:
  - For Add-In card testing: PCI Express Compliance Base Board (CBB2.0) from PCI-SIG
  - For System Board testing: PCI Express Compliance Load Board (CLB2.0) from PCI-SIG
- The free-of-charge R&S ScopeSuite software, which can be installed on a computer or directly on the oscilloscope.

21.6 DDR3 compliance tests (option R&S RTO6-K91)

The option R&S RTO6-K91 enables the automatic processing of DDR3 compliance tests.

The DDR3 compliance test procedures are described in a separate manual:

- R&S®RTO-K91, R&S®RTP-K91 DDR3 Compliance Test, User Manual"

21.6.1 Test equipment

For DDR3 compliance tests, the following test equipment is needed:

- R&S RTP with 4 channels and minimum 8 GHz bandwidth, or
  R&S RTO2000 with 4 channels and minimum 4 GHz bandwidth
  R&S RTO6 with 4 channels and minimum 4 GHz bandwidth
- R&S RTO6-K91 DDR3 compliance test option (required option, installed on the R&S RTO6)
- 4 modular probes, R&S ZM90 with 9 GHz bandwidth. See also Chapter 21.6.1.1, "Soldering guide for modular probes", on page 1222.
- 4 modular probe tips R&S RT-ZMA10
- The free-of-charge R&S ScopeSuite software, which can be installed on a computer or directly on the R&S RTO6.

21.6.1.1 Soldering guide for modular probes

All single-ended signals such as ADD, CMD, DQ, DM, CS, CKE should be soldered in such a way that the signal is connected to the + part on the probe tip. The ground of the signal is connected to the - part on the same probe tip.

All differential signals such as CK, DQS should be soldered differentially:

- The + signal is connected to the + part on the probe tip.
- The - signal is connected to the - part on the same probe tip.
- The ground of the signal is connected to the ground of the same probe tip.

22 Network operation and remote control

The operating system is the basis of the instrument's firmware, it provides basic functions such as logon, password protection, virus protection, and connection to a network.

In addition to working with the R&S RTO6 directly, using the touchscreen and the keys, you can also operate the instrument from a remote PC. Various methods of remote control are supported:

- Using the web interface in a LAN network
- Using Remote Desktop Connection in a LAN network
- Remote control with SCPI commands

- Operating system....1223
- Setting up a network (LAN) connection....1228
- Web interface....1233
- Remote Desktop connection....1238
- Remote control....1240

22.1 Operating system

The R&S RTO6 has a Windows 10 operating system.

The operating system has been configured according to the instrument's features and needs. To ensure that the instrument software functions properly, certain rules must be observed when using the operating system.

NOTICE

Risk of rendering instrument unusable

The instrument is equipped with the Windows operating system. You can install additional software on the instrument, however, additional software can impair instrument function. Thus, run only programs that Rohde & Schwarz has tested for compatibility with the instrument software.

The drivers and programs used on the instrument under Windows are adapted to the instrument. Only install update software released by Rohde & Schwarz to modify existing instrument software.

Changes in the system setup are only required if the network configuration does not comply with the default settings (see Chapter 22.2.1, "Connecting the instrument to the network", on page 1228).

22.1.1 Logon

Windows requires that users identify themselves by entering a user name and password in a logon window.

If the instrument is connected to the network, you are automatically logged on to the network when you log on to the operating system. As a prerequisite, the user name and the password must be identical under Windows and on the network. The instrument provides an auto-logon function that can be configured for standard user and administrator access. The configuration requires the user name and password.

See also: "Log on as" on page 71.

The R&S RTO6 provides two user accounts:

- "Instrument": an administrator account with unrestricted access to the computer/domain (default user)
- "NormalUser": a standard user account with limited access

For both users, the initial password is "894129". It is recommended that you change the password for both users after initial logon. You can change the password in the Windows configuration :

"Start" menu > "Settings" > "Accounts" > "Sign-in options".

Make sure to change the password also in the auto-logon of the R&S RTO6, see below.

By default, the instrument logs on with administrator account and standard password. To restrict the access to system functions, you can change the autologon to "NormalUser".

The following tasks require administrator rights:

- Configuration of a LAN settings, network connection, and firewall settings
- Date and time setup
- Printer installation
- Firmware update
- BIOS update
- Using the web browser
- Touchscreen calibration
- Installation of options
- Moving portable licenses

To configure the auto-logon for a standard user

Default situation: the auto-logon is configured for the administrator ("Instrument").

1. Open "Menu" > "Settings" > "System" > "Users"
2. Set "Logon as" to "User autologon".
3. Enter the "User name": *NormalUser*.
4. Enter the password of the standard user.

5. Restart the instrument.

To configure the auto-logon for administrator

Starting situation: the auto-logon is configured for a standard user ("NormalUser").

1. Open "Menu" > "Settings" > "System" > "Users"
2. Set "Logon as" to "None".
3. Exit the firmware.
4. Sign out from the operating system.
5. Log on as administrator ("Instrument").
6. Set the "Logon as" to "Admin autologon". Enter the "User name": *Instrument* and the administrator's password.
7. Restart the instrument.

22.1.2 Service packs and updates

Microsoft regularly creates security updates and other patches to protect Windows-based operating systems. These are released through the Microsoft Update website and associated update server. Instruments using Windows, especially those that connect to a network, should be updated regularly.

For details and recommendations, see the following Rohde & Schwarz white paper:

- 1EF96: Malware Protection Windows 10

22.1.3 Virus protection

Take appropriate steps to protect your instruments from infection. Use strong firewall settings and scan any removable storage device used with a Rohde & Schwarz instrument regularly. It is also recommended that you install anti-virus software on the instrument. Rohde & Schwarz does NOT recommend running anti-virus software in the background ("on-access" mode) on Windows-based instruments, due to potentially degrading instrument performance. However, Rohde & Schwarz does recommend running it during non-critical hours.

For details and recommendations, see the following Rohde & Schwarz white paper:

- 1EF96: Malware Protection Windows 10

22.1.4 Backup and recovery

The R&S RTO6 has a backup partition with an application for backup and restore the firmware. The backup of the factory default state is always available. You can save additional backups, for example, before a firmware update or to provide different system configurations for different environments.

The backup saves the current instrument installation and its configuration. Data are not saved in the backup.

To save a system backup

1. Before starting the backup, unplug the LAN cable and USB storage devices to minimize the risk of computer virus attacks.
2. Restart the R&S RTO6.

   The boot screen is displayed for about 2 seconds.
3. When the boot screen is displayed, immediately select "Backup", and press [ENTER].
4. Select "Create Backup".
5. Enter a name for the backup and the date. If necessary, you can add information to the description. By default, the description contains the current version of the firmware.
6. Select "Start Backup".

   A progress information dialog is displayed. You can terminate at any time with "Cancel". After the process has been finished, the dialog is closed automatically, and the main dialog is displayed again.
7. In the main dialog, select "Exit and Reboot".

   The backup and restore application is closed, and the R&S RTO6 is restarted.

To restore a backup version

When restoring a backup, the Windows operating system and installed anti-malware software are probably outdated. To minimize the risk of malware threats after restoring a backup, verify and adjust the "Windows Update" settings. Follow the recommendations from Rohde & Schwarz applicable to your instrument. Also, install all Windows security updates that have been published in the meanwhile.

1. Before starting the recovery, unplug the LAN cable and USB storage devices to minimize the risk of computer virus attacks.
2. Restart the R&S RTO6.

   The boot screen is displayed for about 2 seconds.
3. When the boot screen is displayed, immediately select "Backup", and press [ENTER].
4. Select the backup that you want to restore.
5. Select "Restore Selected".
6. Confirm with "Yes".

A progress information dialog is displayed. You can terminate at any time with "Cancel".

During the restore process, the system partition is deleted, formatted and written newly. The data partition is not affected. After the process has been finished, the application is closed automatically, and the R&S RTO6 is restarted.

To delete a backup version

To provide space for new backups, you can remove older backups. The factory default cannot be deleted.

1. Restart the R&S RTO6.
2. When the boot screen is displayed, immediately select "Backup", and press [ENTER].
3. Select the backup that you want to delete.
4. Select "Delete Selected".
5. Confirm with "Yes".
6. In the main dialog, select "Exit and Reboot", or continue with "To save a system backup" on page 1226.

22.1.5 Accessing windows functionality

All required Windows settings can be changed using the touchscreen and the on-screen keyboard that is part of the Windows system. However, modification is easier if you connect a mouse and/or keyboard to the instrument.

To access Windows

► Open "Menu" > 🖵.

The application is minimized to the taskbar and the "Start" menu becomes available.

To access Windows using an external keyboard

1. To open the "Start" menu, press the Windows key or the CTRL + ESC key combination on your keyboard.
2. To access the desktop, press the Windows key + D on your keyboard.

22.2 Setting up a network (LAN) connection

Network environment

Before connecting the product to a local area network (LAN), consider the following:

- Install the latest firmware to reduce security risks.
- For internet or remote access, use secured connections if applicable.
- Ensure that the network settings comply with the security policies of your company. Contact your local system administrator or IT department before connecting your product to your company LAN.
- When connected to the LAN, the product may potentially be accessed from the internet, which may be a security risk. For example, attackers might misuse or damage the product.

Applications of LAN

The LAN connection settings can be configured directly in the Windows operating system. Provided the network administrator has assigned you the appropriate rights and adapted the Windows firewall configuration, you can use the interface, for example:

- To transfer data between a controlling device and the test device, e.g. to run a remote control program.
- To access or control the measurement from a remote computer using the "Remote Desktop" application (or a similar tool)
- To transfer data from a remote computer and back, e.g. using network folders

This section describes how to configure the LAN interface. It includes the following topics:

- Chapter 22.2.1, "Connecting the instrument to the network", on page 1228
- Chapter 22.2.2, "Assigning the IP address", on page 1229

Note that only user accounts with administrator rights can configure LAN networks.

22.2.1 Connecting the instrument to the network

There are two methods to establish a LAN connection to the instrument:

- A non-dedicated network (Ethernet) connection from the instrument to an existing network.
- A dedicated network connection (Point-to-point connection) between the instrument and a single computer.

NOTICE

Risk of network failure

Consult your network administrator before performing the following tasks:

- Connecting the instrument to the network
- Configuring the network
- Changing IP addresses
- Exchanging hardware

Errors can affect the entire network.

► To establish a non-dedicated network connection, connect a commercial RJ-45 cable to one of the LAN ports.
To establish a dedicated connection, connect a (crossover) RJ-45 cable between the instrument and a single PC.

If the instrument is connected to the LAN, Windows automatically detects the network connection and activates the required drivers.

The network card can be operated with a 10/100/1000 Mbps Ethernet IEEE 802.3u interface.

22.2.2 Assigning the IP address

Depending on the network capacities, the TCP/IP address information for the instrument can be obtained in different ways.

- If the network supports dynamic TCP/IP configuration using the Dynamic Host Configuration Protocol (DHCP), all address information can be assigned automatically.
- If the network does not support DHCP, or if the instrument is set to use alternate TCP/IP configuration, the addresses must be set manually.

By default, the instrument is configured to use dynamic TCP/IP configuration and obtain all address information automatically. This means that it is safe to establish a physical connection to the LAN without any previous instrument configuration.

NOTICE

Risk of network errors

Connection errors can affect the entire network. If your network does not support DHCP, or if you choose to disable dynamic TCP/IP configuration, you must assign valid address information before connecting the instrument to the LAN. Contact your network administrator to obtain a valid IP address.

Assigning the IP address on the instrument

1. Open "Menu" > "Settings".

2. Select the "System" tab.
3. In the "Network" tab, select "Advanced".
4. Select "Ethernet".
5. Select "Properties".
6. On the "Networking" tab, select "Internet Protocol Version 4 (TCP/IPv4)", and then select "Properties".
7. Select "Use the following IP address".
8. Enter the address information as obtained from the network administrator.

22.2.3 Using computer names

In a LAN that uses a DNS server (Domain Name System server), each PC or instrument in the LAN can be accessed via an unambiguous device name (or host name, computer name) instead of the IP address. The DNS server translates the host name to the IP address. Device names are useful when a DHCP server is used, as a new IP address can be assigned each time the instrument is restarted.

Each instrument is delivered with an assigned computer name, but this name can be changed.

The default instrument name is a non-case-sensitive string with the following syntax:

<Type><variant>-<serial_number>

The serial number can be found on the rear panel of the instrument. It is the third part of the device ID printed on the bar code sticker:

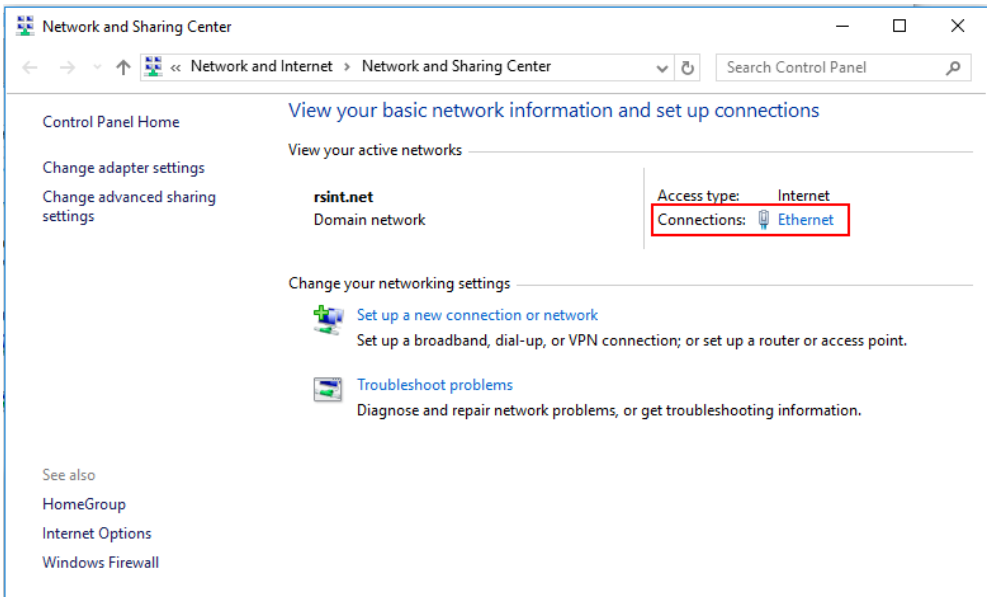
To change the device name

1. Open the "Menu" > "Settings" dialog.
2. Open the "System" dialog.
3. Select the "Network" tab.
4. Enter the new "Device name" and confirm the entry.

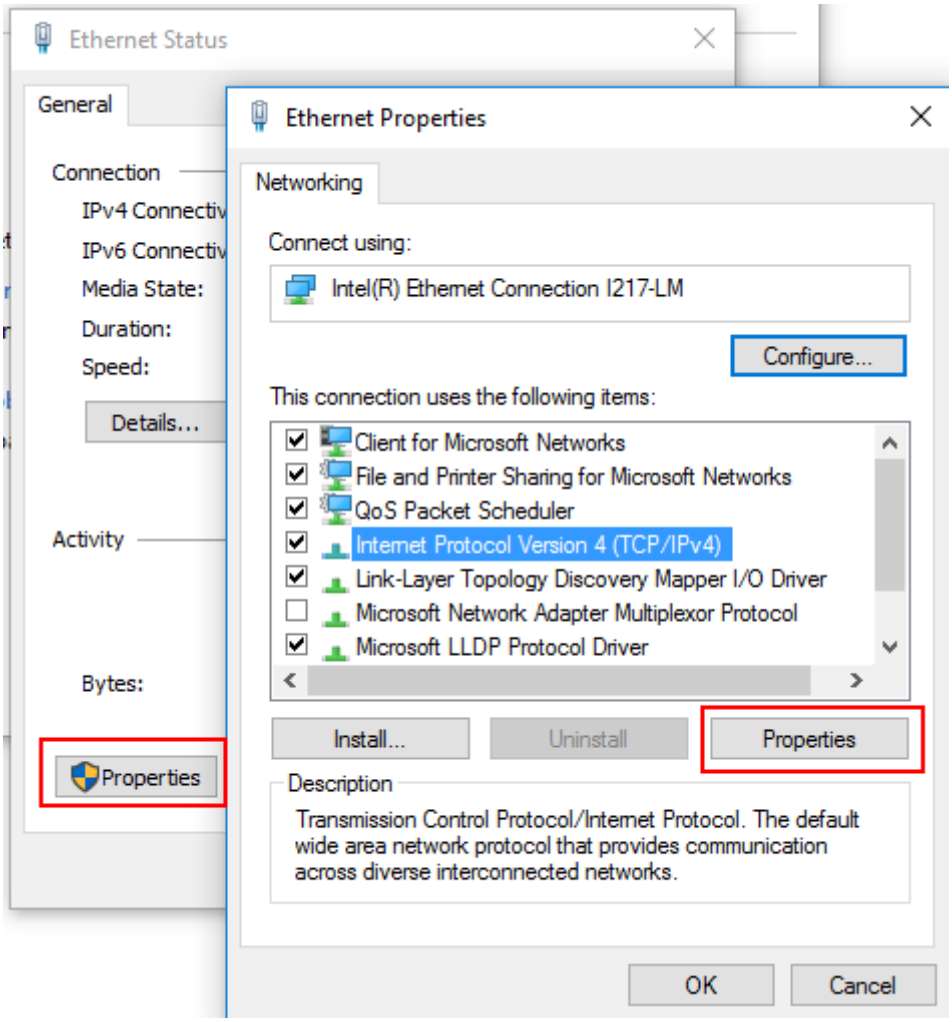
Using a DNS server to determine the IP address

If a DNS server is configured on the R&S RTO6, the server can determine the current IP address for the connection using the permanent computer name.

1. Obtain the name of your DNS domain and the IP addresses of the DNS and WINS servers on your network.
2. Open the "Menu" > "Settings" dialog.
3. Select the "System" tab.
4. Tap "Network".
5. Select "Ethernet".



6. On the "Networking" tab, select "Internet Protocol Version 4 (TCP/IPv4)", and then select "Properties".



7. On the "General" tab, select "Use the following DNS server addresses" and enter your own DNS addresses.

   For more information, refer to the Windows operating system help.

22.2.4 Changing the windows firewall settings

A firewall protects an instrument by preventing unauthorized users from gaining access to it through a network. Rohde & Schwarz highly recommends the use of the firewall on your instrument. Rohde & Schwarz instruments are shipped with the Windows firewall enabled and preconfigured in such a way that all ports and connections for remote control are enabled.

For more details on firewall configuration, see the following Rohde & Schwarz White Paper:

- 1EF96: Malware Protection Windows 10

Note that changing firewall settings requires administrator rights.

22.3 Web interface

If the R&S RTO6 is connected to a computer via LAN, you can operate the instrument from a computer. No additional tools are required, you need only a web browser.

22.3.1 Settings on the R&S RTO6

Access: "Menu" > "Settings" > "System" > "Network"

The LAN status icon on the toolbar of the R&S RTO6 indicates the status of the LAN connection. A green icon indicates that the instrument is connected to the LAN; a red symbol indicates an error - mostly the LAN cable is not connected.

The "Network" tab of the "System" dialog box provides network information. See Chapter 4.1.2, "Network settings", on page 68.

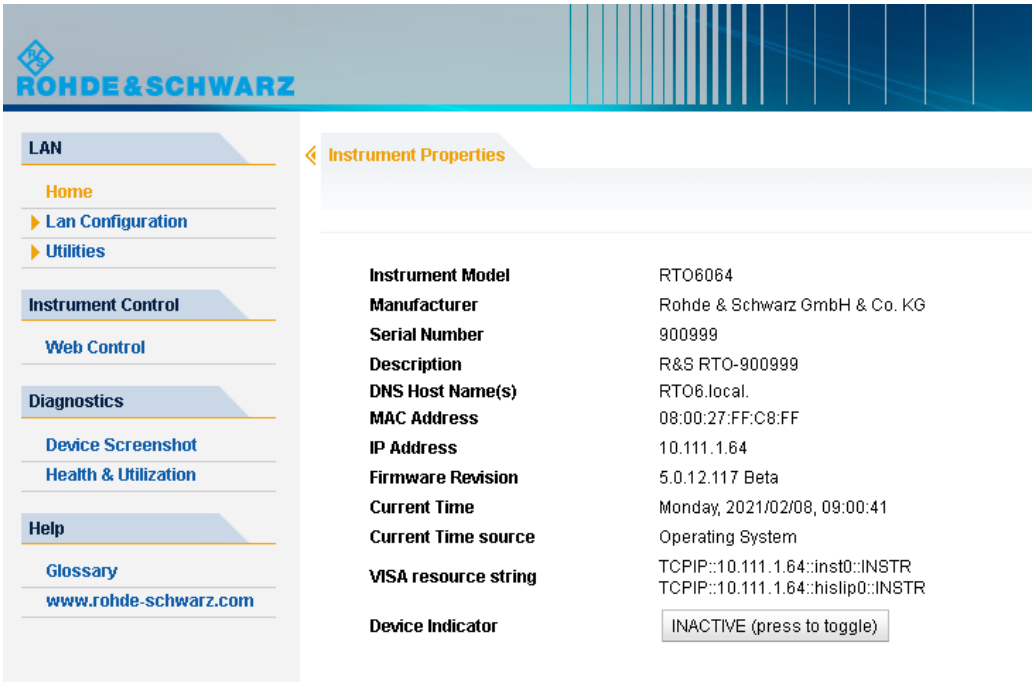
Only users with administrator rights can change LAN settings.

22.3.2 Web browser

The instrument's web interface works with all W3C compliant browsers. Only users with administrator rights on the instrument can use the web interface.

1. Open a web browser on the computer.
2. Type the instrument's host name or IP address in the address field of the browser on your PC, e.g. "http://10.113.10.203".

   The instrument home page opens.



The instrument home page displays the device information required by the LAN standard including the VISA resource string in read-only format.

► To update the "Host Name", press the "Device Indicator" button. The "Device Indicator" is not password-protected.

The most important items in the navigation menu of the browser interface are the following:

- "LAN Configuration" opens the menu with configuration pages.
- "Utilities > Logging" shows log messages and provides buttons to download or clear the logfile.
- "Web Control" emulates the front panel and shows the instrument display. You see a live image of the instrument, and you can operate the instrument remotely. You can use the keys, the knobs and the menus in the same way as directly on the instrument.
- "Device Screenshot": creates a screenshot of the display.
- "Health & Utilization": not yet supported
- "Help" provides a glossary of terms related to the standard, and a link to the Rohde & Schwarz Internet site.

22.3.2.1 LAN configuration

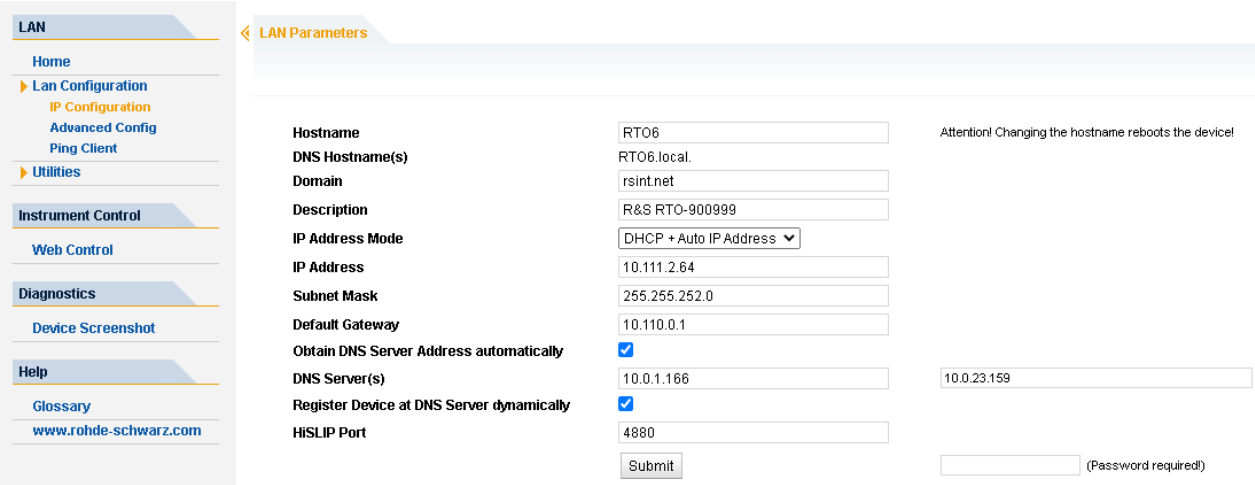
The LAN configuration consists of three parts:

- "IP configuration" provides all mandatory LAN parameters.
- "Advanced LAN Configuration" provides LAN settings that are not declared mandatory by the standard.

- "Ping Client" provides the ping utility to verify the connection between the instrument and other devices.

IP configuration

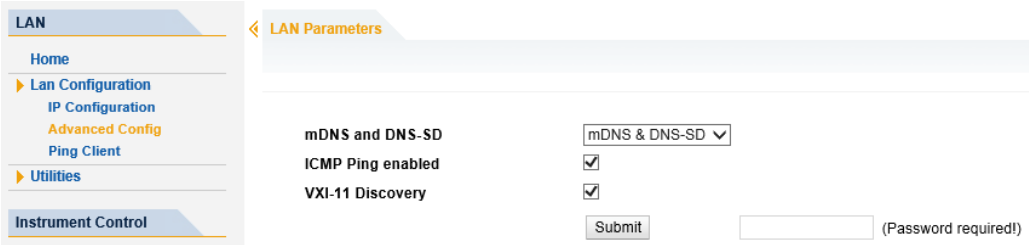
The "LAN Configuration > IP configuration" web page displays all mandatory LAN parameters and allows their modification.



The "TCP/IP Mode" configuration field controls how the IP address for the instrument gets assigned (see also Chapter 22.2.2, "Assigning the IP address", on page 1229). For the manual configuration mode, the static IP address, subnet mask, and default gateway are used to configure the LAN. The automatic configuration mode uses DHCP server or Dynamic Link Local Addressing (automatic IP) to obtain the instrument IP address.

Changing the LAN configuration is password protected. The password is *LxiWebIfc* (notice upper and lower case characters). This password cannot be changed in the current firmware version.

Advanced config



The "LAN Configuration > Advanced Config" parameters are used as follows:

- mDNS and DNS-SD are two additional protocols: Multicast DNS and DNS Service Discovery. They are used for device communication in zero configuration networks working without DNS and DHCP
- "ICMP Ping" must be enabled to use the ping utility.

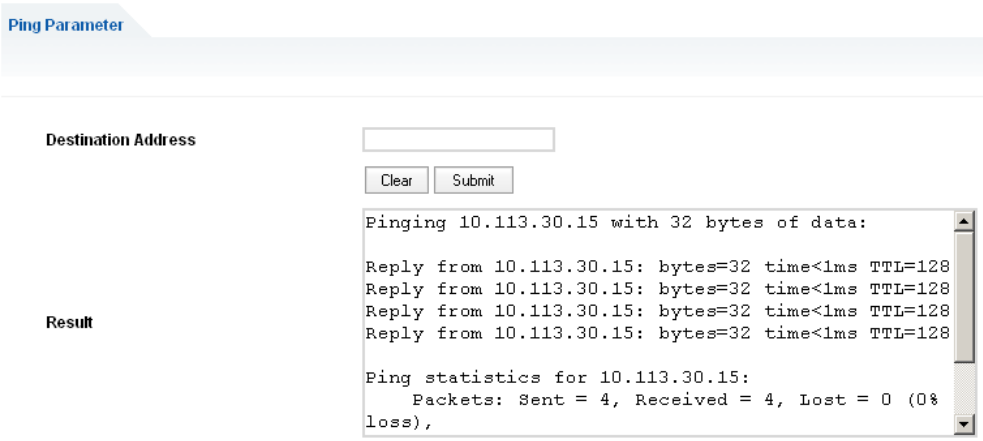
- "VXI-11" is the protocol that is used to detect the instrument in the LAN.

Ping client

Ping is a utility that verifies the connection between the instrument and another device. The ping command uses the `ICMP` echo request and echo reply packets to determine whether the LAN connection is functional. Ping is useful for diagnosing IP network or router failures. The ping utility is not password-protected.

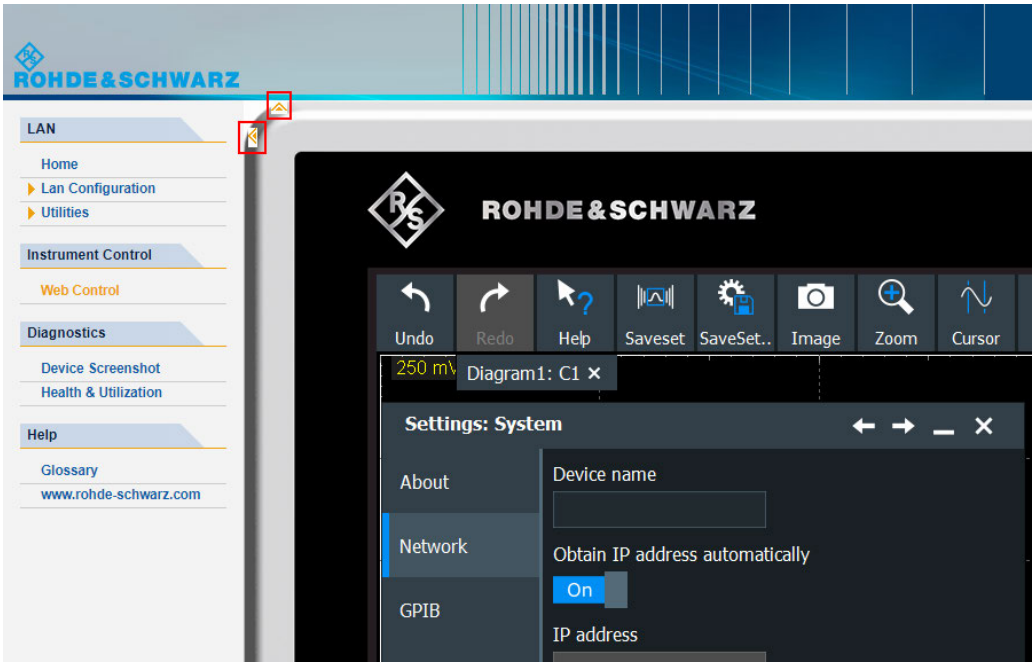
To initiate a ping between the compliant instrument and a second connected device:

1. Enable "ICMP Ping" on the "Advanced Config" page (enabled by default).
2. On the "Ping Client" page, enter the IP address of the second device **without the ping command and without any further parameters** into the "Destination Address" field (e.g. *10.113.10.203*).
3. Click "Submit".



22.3.2.2 Web control

"Web Control" emulates the front panel and shows the instrument display. You see a live image of the instrument, and you can operate the instrument remotely. You can use the keys, the knobs and the menus in the same way as directly on the instrument. The Web control replaces VNC as control tool for remote operation.

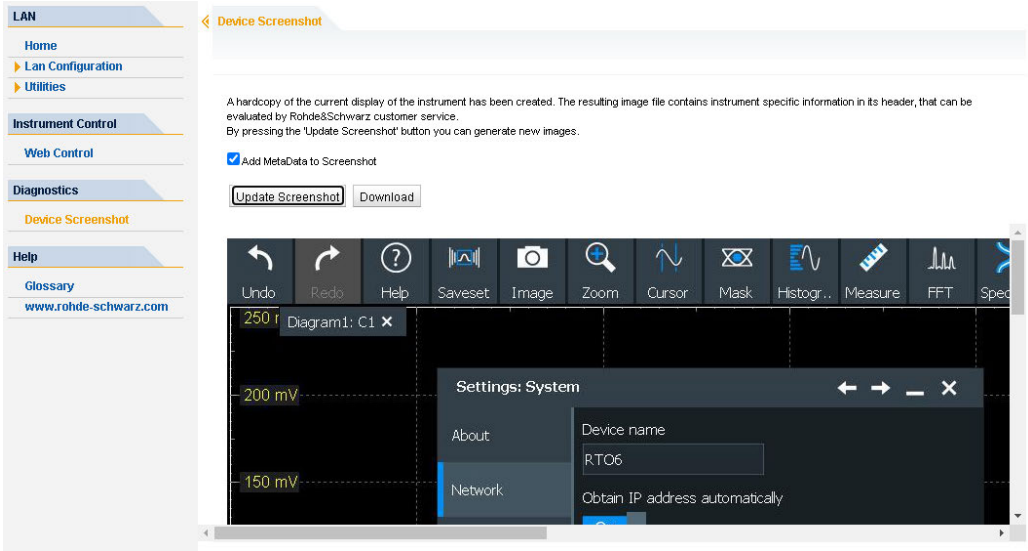


In the upper left corner of the emulated display, you see two arrows:

- The up/down arrow hides or shows the front panel.
- The left/right arrow hides or shows the menu.

22.3.2.3 Device screenshot

If you click "Device Screenshot", a screenshot of the current instrument display is shown on the computer.



Add MetaData to Screenshot
Adds the instrument information to PNG and JPG files. Meta information is saved as EXIF information and can be read, for example, using the ExifTool, see "Meta information in screenshots" on page 475.

Update Screenshot
Updates the display.

Download
Saves the screenshot to the download directory of the computer. By default, JPG format is saved.

To save the screenshot in PNG format, select "Save as", select "All files" as type, and enter the filename with extension *.png*.

22.4 Remote Desktop connection

Remote Desktop is a Windows application, which can be used to access and control the instrument from a remote computer through a LAN connection. While the instrument is in operation, the contents of the instrument screen are displayed on the remote computer. Remote Desktop provides access to all applications, files, and network resources of the instrument. Thus, remote operation of the instrument is possible.

NOTICE

Risk of unauthorized access

If you enable the Windows Remote Desktop application on the instrument, anyone using the network who knows the computer name and login data can access it. To prevent unauthorized access, make sure that the Remote Desktop application on the instrument is disabled: "Start" > "Settings" > "Control Panel" > " System"

To set up a Remote Desktop connection

1. Enable remote desktop control on the instrument. See "Enabling remote desktop control on the instrument" on page 1238 for details.
2. Connect the instrument and the remote computer to a LAN. See Chapter 22.2.1, "Connecting the instrument to the network", on page 1228 for details.
3. Set up the Remote Desktop connection between the remote computer and the instrument.

Remote Desktop Connection is part of the operating system and can be accessed via "Start > Windows Accessories > Remote Desktop Connection.".

Enabling remote desktop control on the instrument

1. Press the [Setup] key and select the "System" tab.

2. Tap "System".
3. Select "Remote settings".

   The "Remote" tab of the "System Properties" is shown.
4. Under "Remote Desktop", activate "Allow remote connections to this computer".

   Note: Remote Desktop access and firewall settings.
   When you enable or disable the Windows Remote Desktop option (in the "System Properties"), the associated firewall settings are adapted automatically.
5. If necessary, click "Select Users" and select users who are allowed to access the R&S RTO6 via Remote Desktop.

   The user account under which configuration is carried out is automatically enabled for Remote Desktop.

Setting up the Remote Desktop connection on the remote computer

1. On the remote computer, select "Start > Windows Accessories > Remote Desktop Connection".
2. Enter the instrument's name or IP address in the dialog box.
   See also Chapter 22.2.2, "Assigning the IP address", on page 1229.
3. Enter the user ID and password for the instrument. See Chapter 22.1, "Operating system", on page 1223 for details.
4. Click "Connect".

   When the connection has been set up, the instrument's screen appears on the remote computer.

For detailed information about Remote Desktop and the connection, refer to the Windows Help.

Helpful settings for Remote Desktop

The following settings for the Remote Desktop connection can make working on the remote PC more convenient.

1. When setting up the connection to the instrument, you can configure the connection settings in the "Remote Desktop Connection" dialog box.
   Click the "Show Options".

   The dialog box is expanded to display the configuration data.
2. Customize the settings:
   a) On the "Experience" tab, select the appropriate connection to optimize the connection speed.

b) On the "Local Resources" tab:
   - To use printers connected to the remote PC while accessing them from the instrument, activate "Printers" under "Local devices and resources".
   - If you want access drives of the remote PC from the instrument, e.g. to store settings or to copy files from the PC to the instrument:
     – Select "More" under "Local devices and resources".
     – Select the "Drives" that are needed.
       Windows maps the selected drives of the remote PC to corresponding network drives. When a connection is established, a warning on the PC indicates that the drives are enabled for access from the instrument.

c) On the "Display" tab:
   - Use the slider to set the size of the R&S RTO6 window on the remote PC desktop.
   - Activate "Display the connection bar when in full screen mode".
     A bar with the network address of the instrument is displayed on the screen, which you can use to reduce, minimize or close the window.

d) On the "General" tab, you can save the connection settings for later use: click "Save As".

Terminating Remote Desktop Control

A Remote Desktop connection can be terminated either on the R&S RTO6 or on the remote PC. The connection can be established again any time as long as remote control is enabled on the instrument. Consider the notice "Risk of unauthorized access" on page 1238.

► To terminate the connection on the remote PC, close the "Remote Desktop" window, or select "Start > Disconnect".

22.5 Remote control

Remote control automates the operation of the instrument using SCPI commands, scripts and programs.

For general information on remote control of Rohde & Schwarz products via SCPI, refer to www.rohde-schwarz.com/rc-via-scpi.

The following sections describe the instrument-specific basics of remote control. SCPI commands are listed described in chapter "Remote Control Commands".

- Remote control interfaces and protocols....1241
- Remote settings....1242
- Starting and stopping remote control....1244
- Remote control - status reporting system....1245

22.5.1 Remote control interfaces and protocols

The instrument supports different interfaces for remote control. The following table gives an overview.

Table 22-1: Remote control interfaces and protocols

Interface	Protocols, VISA address string	Remarks

Local Area Network (LAN)	**Protocol HiSLIP**
	VISA address string:
	`TCPIP::<host address>::hislip0[,<port>][::INSTR]`
	Protocol VXI-11
	VISA address string:
	`TCPIP::<host address>[::inst0]::[INSTR]`
		The LAN connector is located on rear panel of the instrument.
		The interface is based on TCP/IP and supports various protocols.

GPIB (IEC/IEEE Bus Interface)	VISA address string: GPIB::primary address[::INSTR] (no secondary address)	The GPIB bus interface according to standard IEC 625.1/IEEE 488.1 is located on the rear panel of the instrument. The GPIB bus interface requires option R&S RTO6-B10.

Within this interface description, the term GPIB is used as a synonym for the IEC/IEEE bus interface.

22.5.1.1 LAN interface

To be integrated in a LAN, the instrument is equipped with a LAN interface, consisting of a connector, a network interface card and protocols. The interface details are given in the specifications.

For remote control via a network, the PC and the instrument must be connected via the LAN interface to a common network with TCP/IP network protocol. They are connected using a commercial RJ45 cable (shielded or unshielded twisted pair category 5). The TCP/IP network protocol and the associated network services are preconfigured on the instrument. Software for instrument control and the VISA program library must be installed on the controller.

IP address

Only the IP address or a valid DNS host name is required to set up the connection. The host address is part of the "VISA resource string" used by programs to identify and control the instrument. The VISA resource string has the form:

`TCPIP::<host address>::hislip0[,<port>][::INSTR]` for HiSLIP protocol

`TCPIP::<host address>[::inst0][::INSTR]` for VXI-11 protocol

Where:

- `host address` identifies the instrument in the network, usually the IP address. If the LAN is supported by a DNS server, the host name can be used instead of the IP address. The DNS server (Domain Name System server) translates the host name to the IP address.

- `hislip0` indicates the HiSLIP protocol
- `inst0` is the default LAN device name. VISA supports several devices running on the instrument. On R&S RTO6, only one device is configured, so the LAN device name can be omitted.
- `INSTR` specifies a VISA resource of the type INSTR. By default, the VISA resource name control is set to the INSTR class.

Example: HiSLIP

IP address is *192.1.2.3*: the valid resource string is: `TCPIP::192.1.2.3::hislip0`

Instrument name is *RSRT1*: the valid resource string is: `TCPIP::RSRT1::hislip0`.

DNS host name is *RTO6-123456*: the valid resource string is: `TCPIP::RTO6-123456::hislip0`.

Example: VXI-11

IP address is *192.1.2.3*: the valid resource string is: `TCPIP::192.1.2.3`

Instrument name is *RSRT1*: the valid resource string is: `TCPIP::RSRT1`.

DNS host name is *RTO6-123456*: the valid resource string is: `TCPIP::RTO6-123456`.

See also:

- Find IP address: "Menu" > "Settings" > "System" tab, see "IP Address" on page 69
- Chapter 22.2.2, "Assigning the IP address", on page 1229

22.5.1.2 GPIB interface (IEC/IEEE bus interface)

A GPIB interface is provided with option R&S RTO6-B10.

To control the instrument remotely via GPIB, the instrument and the controlle must be connected by a GPIB bus cable. The total cable length is restricted to a maximum of 15 m; the cable length between two instruments should not exceed 2m.

GPIB instrument address

The GPIB address is set on the instrument, see Chapter 22.5.2.1, "GPIB settings", on page 1242.

22.5.2 Remote settings

The settings on this tab are required for remote control of the instrument via a connected computer.

22.5.2.1 GPIB settings

Access: "Menu" > "Settings" > "System" > "Remote"

Displays the GPIB settings of the instrument.

It requires an optional GPIB bus card, option R&S RTO6-B10.

Address
Indicates the GPIB address of the instrument if an optional GPIB bus card is installed.

The remote control address is factory-set to 20, but it can be changed if it does not fit in the network environment. For remote control, addresses 0 through 30 are allowed. The GPIB address is maintained after a reset of the instrument settings. Be aware that changing the address has major effects on the communication to the remote computer.

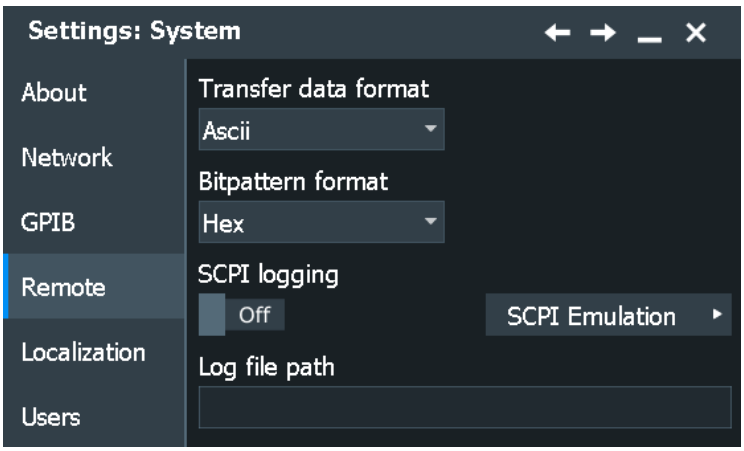
Remote command:
`GPIB:ADDRess` on page 1294

Terminator
Specifies the symbol that is used as a terminator in GPIB communication.

Remote command:
`GPIB:TERMinator` on page 1295

22.5.2.2 Remote settings

Access: "Menu" > "Settings" > "System" > "Remote"



The following settings are required for remote control of the instrument via a connected computer.

Transfer data format
Selects the data format that is used for transmission of waveform data from the instrument to the controlling computer.

Waveform data can be retrieved using the following commands:
- `CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?`
- `CALCulate:MATH<m>:DATA[:VALues]?`
- `REFCurve<m>:DATA[:VALues]?`
- `DIGital<m>:DATA[:VALues]?`

The content of the data stream can be defined with "Menu" > "Save/Recall" > "Save > Waveforms > Interleaved X/Y" (or `EXPort:WAVeform:INCXvalues`).

"Ascii" Data values are returned in ASCII format as a list of comma separated values in floating point format.

"FLOAT"	Binary format. The data is stored as binary data (Definite Length Block Data according to IEEE 488.2).
"INT8"	Signed integer data with length 8 bit.
"INT16"	Signed integer data with length 16 bit. The Byte order can be set using . For details on the formats, refer to the description of the remote command.

Remote command:
`FORMat[:DATA]` on page 1292

Byte order
Sets the endianess for INT16 data:
- LSB first: little endian, least significant byte first
- MSB first: big endian, most significant byte first

Remote command:
`FORMat:BORDer` on page 1293

Bit pattern format
Sets the format for all bit pattern queries.

Remote command:
`FORMat:BPATtern` on page 1294

SCPI logging
If enabled, all received remote commands are written into a text file. Enter the path and filename with extension (log, txt, or csv) in "Log file path".

SCPI emulation
Not available for R&S RTO6

22.5.3 Starting and stopping remote control

22.5.3.1 Starting a remote control session

When you switch on the instrument, it is always in manual operation state ("local" state). It can be operated via the front panel, the touch screen and external keyboard and/or mouse.

► To start remote control:
- Send a command from the controller.
- VXI-11 protocol (LAN or USB interface): Use `&GTR` interface message.

While remote control is active, the instrument settings are optimized for maximum measurement speed; the display is switched off. Operation via the front panel is disabled.

On the touch screen, two buttons appear in the upper left corner: "Local" and "View".

22.5.3.2 Using the display during remote control

You can observe the screen while a remote control script is executed. This is helpful for program test purposes but tends to slow down the measurement. Therefore it is recommended that you switch off the display in real measurement applications where a tested program script is to be executed repeatedly.

► To switch on the display, do one of the following:

- Tap the "View" button in the upper left corner of the touch screen.
- Use the `SYSTem:DISPlay:UPDate ON` command.

► To switch off the display, do one of the following:

- Tap the "View" button again.
- Use the `SYSTem:DISPlay:UPDate OFF` command.

22.5.3.3 Returning to manual operation

The instrument switches back to manual operation when the remote connection is closed. Besides, you can return to manual operation manually or via remote control.

► To return to manual operation:

- Tap the "Local" button in the upper left corner of the touch screen.
- VXI-11 protocol: Use `&GTL` interface message.

22.5.4 Remote control - status reporting system

The status reporting system stores all information on the current operating state of the instrument, and on errors which have occurred. This information is stored in the status registers and in the error queue. Both can be queried via `STATus`... commands.

22.5.4.1 Hierarchy of status registers

As shown in the following figure, the status information is of hierarchical structure.

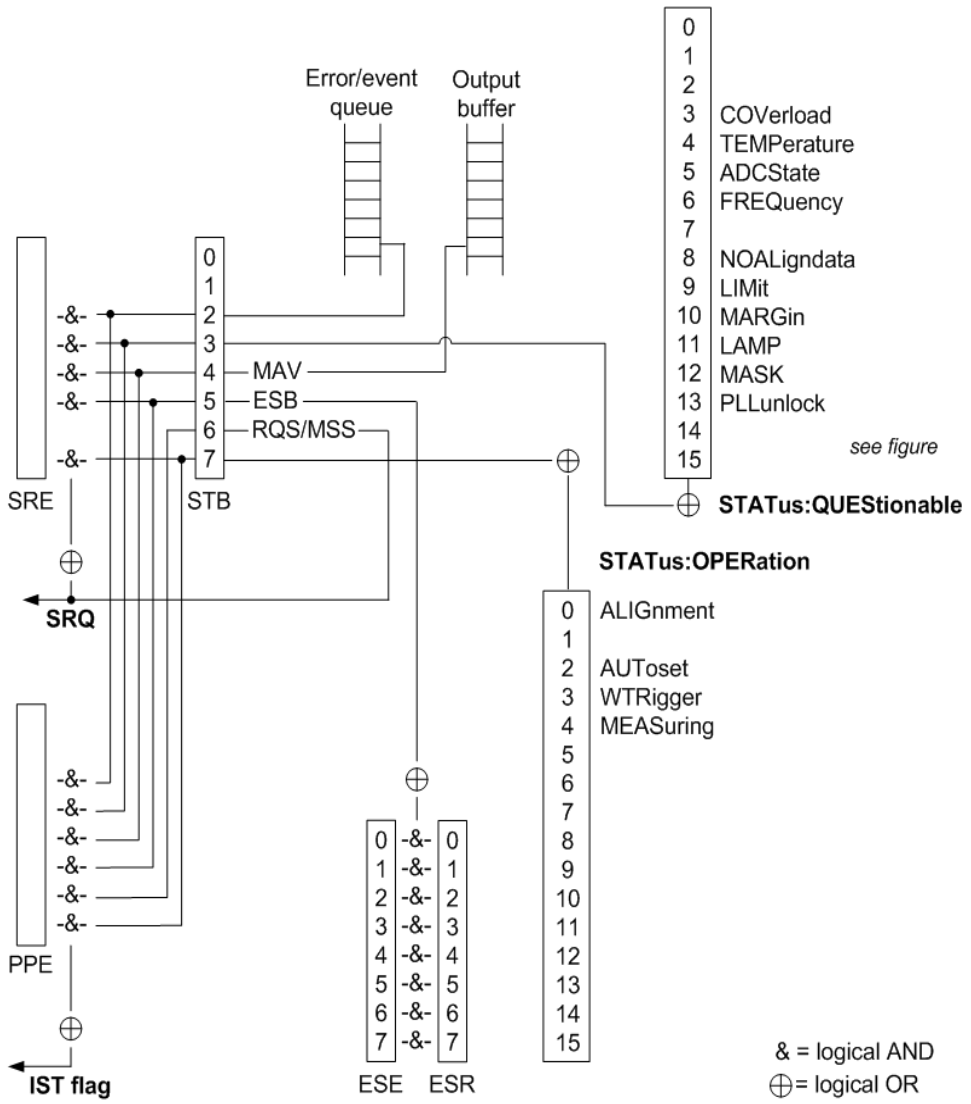


Figure 22-1: Overview of the status registers hierarchy

All status registers have the same internal structure.

22.5.4.2 Contents of the status registers

STATus:OPERation register

In the `CONDition` part, this register contains information on which actions the instrument is executing. In the `EVENt` part, it contains information on which actions the instrument has executed since the last reading.

It can be read using the commands `STATus:OPERation:CONDition?` or `STATus:OPERation[:EVENt]?`, see also Chapter 24.26.1, "STATus:OPERation register", on page 2572.

Table 22-2: Bits in the STATus:OPERation register

Bit No.	Meaning
0	`ALIGnment` This bit is set as long as the instrument is performing a self-alignment.
1	Not used
2	`AUToset` This bit is set while the instrument is performing an auto setup.
3	`WTRIgger` The wait for trigger status bit indicates that the instrument is ready to trigger, and the pre-trigger time is expired. The bit is set if the instrument did not trigger for more than 10 ms. The bit is only valid if the trigger event is initiated by the user, for example, using an external generator. If the instrument triggers on signals or if it is in auto trigger mode, the bit status is undefined.
4	`MEASuring` The bit is set as long as an acquisition - sampling and postprocessing - is running. In run continuous mode, the bit is always set.
5 - 15	Not used

STATus:QUEStionable register

This register contains information about indefinite states which may occur if the unit is operated without meeting the specifications. It can be read using the commands `STATus:QUEStionable:CONDition?` and `STATus:QUEStionable[:EVENt]?`

The remote commands for the STATus:QUEStionable register are described in Chapter 24.26.2, "STATus:QUEStionable registers", on page 2573.

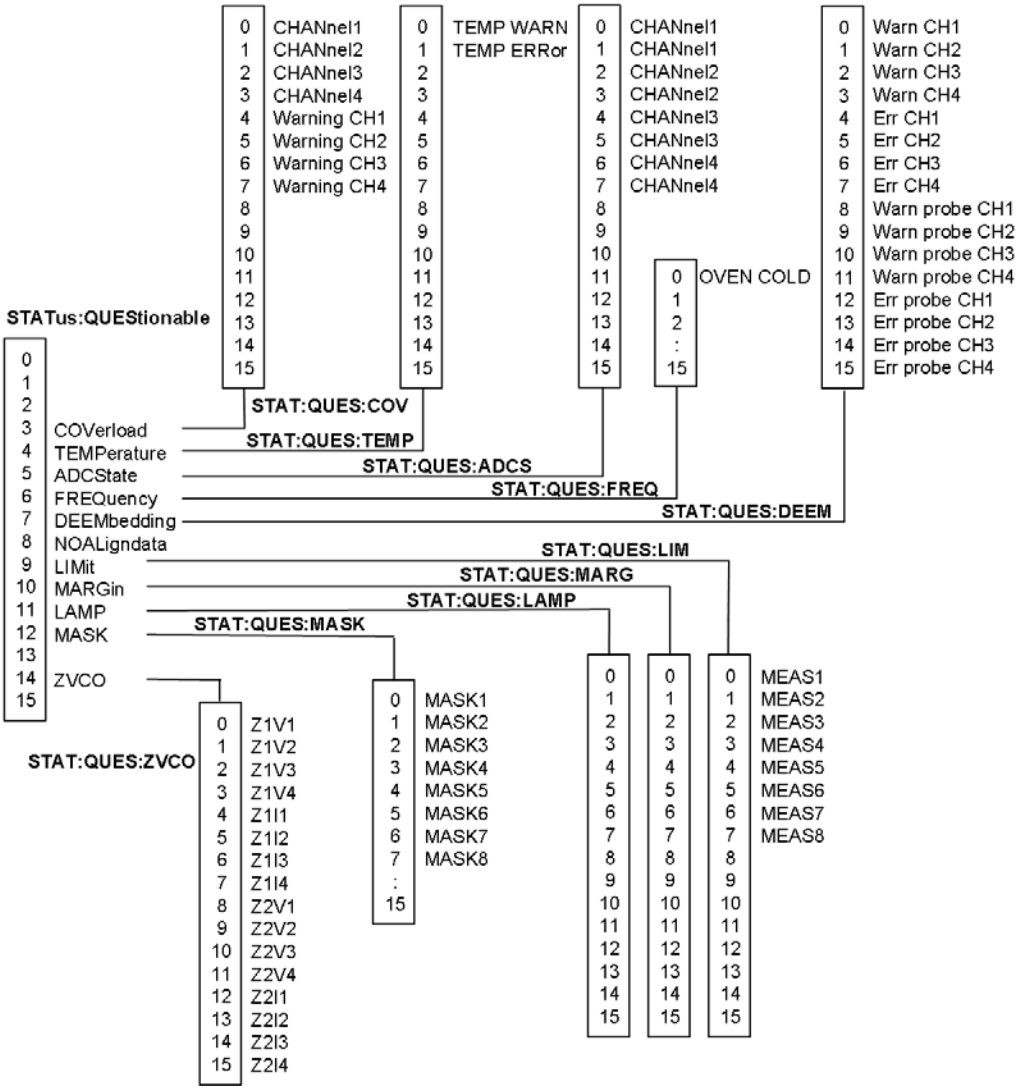


Figure 22-2: Overview of the STATus:QUEStionable register

Table 22-3: Bits in the STATus:QUEStionable register

Bit No.	Meaning
0 to 2	not used
3	`COVerload` This bit is set if a questionable channel overload occurs (see "STATus:QUEStionable:COVerload register" on page 1249).
4	`TEMPerature` This bit is set if a questionable temperature occurs (see "STATus:QUEStionable:TEMPerature register" on page 1250).
5	`ADCState` The bit is set if the signal is clipped on the upper or lower edge of the screen -overflow of the ADC occurs (see "STATus:QUEStionable:ADCState register" on page 1250).

Bit No.	Meaning
6	`FREQuency` This bit is set if there is anything wrong with the frequency of the local oscillator or the reference frequency (see "STATus:QUEStionable:FREQuency register" on page 1250).
7	`DEEMbedding` The bit is set if a warning or an error occurs during deembedding (see "STATus:QUEStionable:DEEMbedding register" on page 1251).
8	`NOALigndata` This bit is set if no alignment data is available - the instrument is uncalibrated.
9	`LIMit` This bit is set if a limit value is violated (see "STATus:QUEStionable:LIMit, STATus:QUEStionable:MARGin registers" on page 1251).
10	`MARGin` This bit is set if a margin value is violated, and the limit value is not violated (see "STATus:QUEStionable:LIMit, STATus:QUEStionable:MARGin registers" on page 1251).
11	`LAMP` (Low AMPlitude) This bit is set if the magnitude of the signal is too low to get reliable measurement results. See "STATus:QUEStionable:LAMP register" on page 1252.
12	`MASK` This bit is set if a mask value is violated (see "STATus:QUEStionable:MASK register" on page 1252
13	Not used
14	`ZVCO`: This bit is set if a channel overload occurs at one or more input channels of the R&S RT-ZVC multi-channel power probe. See "STATus:QUEStionable:ZVCO register" on page 1252.
15	This bit is always 0.

STATus:QUEStionable:COVerload register

This register contains all information about overload of the channels. The bit is set if the assigned channel is overloaded, or if an overload risk occurred (overload warning).

Table 22-4: Bits in the STATus:QUEStionable:COVerload register

Bit No.	Meaning
0	Overload on `CHANnel1`
1	Overload on `CHANnel2`
2	Overload on `CHANnel3`
3	Overload on `CHANnel4`
4	Overload warning for `CHANnel1`
5	Overload warning for `CHANnel2`
6	Overload warning for `CHANnel3`
7	Overload warning for `CHANnel4`

Bit No.	Meaning
8	Overload on external trigger input
9	Overload warning for external trigger input

STATus:QUEStionable:TEMPerature register

This register contains information about the instrument's temperature.

Table 22-5: Bits in the STATus:QUEStionable:TEMPerature register

Bit No.	Meaning
0	`TEMP WARN` This bit is set if a temperature warning on channel 1, 2, 3 or 4 occurred.
1	`TEMP ERRor` This bit is set if a temperature error on channel 1, 2, 3 or 4 occurred.

STATus:QUEStionable:ADCState register

This register contains all information about overflow of the ADC.

The bit is set if the assigned channel signal is clipped on the upper or lower edge of the screen. In this case, the signal does not fit in the range of the ADC and overflow occurs.

Table 22-6: Bits in the STATus:QUEStionable:ADCState register

Bit No.	Meaning
0	`CHANnel1`, clipping on the upper limit
1	`CHANnel1`, clipping on the lower limit
2	`CHANnel2`, clipping on the upper limit
3	`CHANnel2`, clipping on the lower limit
4	`CHANnel3`, clipping on the upper limit
5	`CHANnel3`, clipping on the lower limit
6	`CHANnel4`, clipping on the upper limit
7	`CHANnel4`, clipping on the lower limit

STATus:QUEStionable:FREQuency register

Table 22-7: Bits in the STATus:QUEStionable:FREQuency register

Bit No.	Meaning
0	Oven cold. This bit is set if the reference oscillator has not yet attained its operating temperature.

STATus:QUEStionable:DEEMbedding register

Table 22-8: Bits in the STATus:QUEStionable:DEEMbedding register

Bit No.	Meaning
0 1 2 3	Warning for channel 1, 2, 3, 4 accordingly. The bits are set if there are problems in the deembedding configuration of the assigned channel.
4 5 6 7	Error for channel 1, 2, 3, 4 accordingly. The bits are set if the deembedding configuration of the assigned channel causes an error.
8 9 10 11	Warning for channel 1, 2, 3, 4 accordingly. The bits are set if there are problems in the probe deembedding of the assigned channel.
12 13 14 15	Error for channel 1, 2, 3, 4 accordingly. The bits are set if the probe deembedding of the assigned channel causes an error.

STATus:QUEStionable:LIMit, STATus:QUEStionable:MARGin registers

These registers contain information about the observance of the limits or margins of measurements.

The LIMit bit is set if the limit of the assigned measurement is violated. The MARGin bit is set if the margin of the assigned measurement is violated but not the limit (the limit bit is not set).

The status bits are set if the limits or margins of one or more assigned measurements are violated.

If multiple measurements are active, all measurement results affect the status bits. Statistical results do not change the status bits.

Table 22-9: Bits in the STATus:QUEStionable:LIMit and STATus:QUEStionable.MARGin registers

Bit No.	Meaning
0	`MEAS1`
1	`MEAS2`
2	`MEAS3`
3	`MEAS4`
4	`MEAS5`
5	`MEAS6`

Bit No.	Meaning
6	MEAS7
7	MEAS8

STATus:QUEStionable:LAMP register

The LAMP (Low AMPlitude) bit is set if the magnitude of the signal is too low to get reliable measurement results.

Table 22-10: Bits in the STATus:QUEStionable:LAMP register

Bit No.	Meaning
0	MEAS1
1	MEAS2
2	MEAS3
3	MEAS4
4	MEAS5
5	MEAS6
6	MEAS7
7	MEAS8

STATus:QUEStionable:MASK register

This register contains information about the violation of masks. This bit is set if the assigned mask is violated.

Table 22-11: Bits in the STATus:QUEStionable:MASK register

Bit No.	Meaning
0	MASK1
1	MASK2
2	MASK3
3	MASK4
4	MASK5
5	MASK6
6	MASK7
7	MASK8

STATus:QUEStionable:ZVCO register

This register contains all information about overload of the R&S RT-ZVC input channels. The bit is set if the assigned channel is overloaded.

Table 22-12: Bits in the STATus:QUEStionable:ZVCO register

Bit No.	Meaning
0	Overload on `Z1V1`
1	Overload on `Z1V2`
2	Overload on `Z1V3`
3	Overload on `Z1V4`
4	Overload on `Z1I1`
5	Overload on `Z1I2`
6	Overload on `Z1I3`
7	Overload on `Z1I4`
8	Overload on `Z2V1`
9	Overload on `Z2V2`
10	Overload on `Z2V3`
11	Overload on `Z2V4`
12	Overload on `Z2I1`
13	Overload on `Z2I2`
14	Overload on `Z2I3`
15	Overload on `Z2I4`

22.5.4.3 Reset values of the status reporting system

The following table contains the different commands and events causing the status reporting system to be reset. None of the commands, except `*RST` and `SYSTem:PRESet`, influence the functional instrument settings. In particular, `DCL` does not change the instrument settings.

Table 22-13: Resets of the status reporting system

Event	Switching on supply voltage Power-On-Status-Clear		DCL, SDC (Device Clear, Selected Device Clear)	*RST or SYS-Tem:PRE Set	STA-Tus:PRE-Set	*CLS
Effect	0	1				
Clear STB, ESR	-	yes	-	-	-	yes
Clear SRE, ESE	-	yes	-	-	-	-
Clear PPE	-	yes	-	-	-	-
Clear EVENt parts of the registers	-	yes	-	-	-	yes

Event	Switching on supply voltage Power-On-Status-Clear		DCL, SDC (Device Clear, Selected Device Clear)	*RST or SYS-Tem:PRE Set	STA-Tus:PRE-Set	*CLS
Effect	**0**	**1**				
Clear ENABle parts of all OPERation and QUEStionable registers; Fill ENABle parts of all other registers with "1".	-	yes	-	-	yes	-
Fill PTRansition parts with "1"; Clear NTRansition parts	-	yes	-	-	yes	-
Clear error queue	yes	yes	-	-	-	yes
Clear output buffer	yes	yes	yes	1)	1)	1)
Clear command processing and input buffer	yes	yes	yes	-	-	-
1) The first command in a command line that immediately follows a <PROGRAM MESSAGE TERMINATOR> clears the output buffer.						

23 Maintenance and support

The instrument does not need periodic maintenance. Only the cleaning of the instrument is essential.

To protect the front panel and to transport the instrument to another workplace safely and easily, various accessories are provided. Refer to the data sheet for available covers and cases and their order numbers.

The adjustment of the OCXO oscillator is described in the service manual.

23.1 Cleaning

How to clean the product is described in "Cleaning the product" on page 15.

Do not use any liquids for cleaning. Cleaning agents, solvents, acids and bases can damage the front panel labeling, plastic parts and display.

23.2 Contacting customer support

Technical support – where and when you need it

For quick, expert help with any Rohde & Schwarz product, contact our customer support center. A team of highly qualified engineers provides support and works with you to find a solution to your query on any aspect of the operation, programming or applications of Rohde & Schwarz products.

Contact information

Contact our customer support center at www.rohde-schwarz.com/support, or follow this QR code:

Figure 23-1: QR code to the Rohde & Schwarz support page

23.3 Information for technical support

If you encounter problems that you cannot solve yourself, contact your Rohde & Schwarz support center, see Contacting customer support.

The support center finds solutions more quickly and efficiently, if you provide them with information on the instrument and an error description. To create, collect and save the required information you can use the RTxServiceReporter. The RTxServiceReporter creates a ZIP file with a complete bug report, all relevant setup information, reporting and log files, and the instrument configuration (device footprint).

1. Press the app key on the frontpanel.
2. On the "R&S Apps" tab, tap "Servie Reporter".

   The RTxServiceReporter creates the report and saves it as ZIP file directly on the Windows desktop.
3. Attach the report file to an email in which you describe the problem. Send the email to the customer support address for your region as listed in the internet.

On the instrument, you can find log files, the device footprint and report files, and other information in the "AppData" folder on the Windows desktop (`C:\ProgramData\Rohde-Schwarz\RTx`).

The "Maintenance" dialog box also provides information on your instrument configuration which can be helpful in case you need support.

23.4 Data security

If you have to send the instrument to the service, or if the instrument is used in a secured environment, consider the document "Instrument Security Procedures" that is delivered on the R&S RTO6 web page.

Instrument configuration data and user data are stored on a removable hard disk only. Thus it is sufficient to remove the hard disk before the instrument leaves a secured environment. Details are given in the document mentioned above.

23.5 Transporting

Lifting and carrying

See: "Lifting and carrying the instrument" on page 12

Packing

Use the original packaging material. It consists of antistatic wrap for electrostatic protection and packing material designed for the product.

If you do not have the original packaging, use similar materials that provide the same level of protection.

Securing

When moving the R&S RTO6 in a vehicle or using transporting equipment, make sure that the R&S RTO6 is properly secured. Only use items intended for securing objects.

Transport altitude

The maximum transport altitude without pressure compensation is specified in the data sheet.

23.6 Storage

Protect the product against dust. Ensure that the environmental conditions, e.g. temperature range and climatic load, meet the values specified in the data sheet.

23.7 Disposal

Rohde & Schwarz is committed to making careful, ecologically sound use of natural resources and minimizing the environmental footprint of our products. Help us by disposing of waste in a way that causes minimum environmental impact.

Disposing electrical and electronic equipment

A product that is labeled as follows cannot be disposed of in normal household waste after it has come to the end of its service life. Even disposal via the municipal collection points for waste electrical and electronic equipment is not permitted.

Figure 23-2: Labeling in line with EU directive WEEE

Rohde & Schwarz has developed a disposal concept for the eco-friendly disposal or recycling of waste material. As a manufacturer, Rohde & Schwarz completely fulfills its obligation to take back and dispose of electrical and electronic waste. Contact your local service representative to dispose of the product.

24 Remote control commands

This chapter describes all remote commands available for R&S RTO6 and provides examples and information how to use the commands.

For further information on remote control, see Chapter 22.5, "Remote control", on page 1240.

24.1 Conventions used in remote command description

The following conventions are used in the remote command descriptions:

- **Command usage**
  If not specified otherwise, commands can be used both for setting and for querying parameters.
  If a command can be used for setting or querying only, or if it initiates an event, the usage is stated explicitly.
- **Parameter usage**
  If not specified otherwise, a parameter can be used to set a value and it is the result of a query.
  Parameters required only for setting are indicated as **Setting parameters**.
  Parameters required only to refine a query are indicated as **Query parameters**.
  Parameters that are only returned as the result of a query are indicated as **Return values**.
- **Conformity**
  Commands that are taken from the SCPI standard are indicated as **SCPI confirmed**. All commands used by the R&S RTO6 follow the SCPI syntax rules.
- **Asynchronous commands**
  A command which does not automatically finish executing before the next command starts executing (overlapping command) is indicated as an **Asynchronous command**.
- **Reset values (*RST)**
  Default parameter values that are used directly after resetting the instrument (`*RST` command) are indicated as ***RST** values, if available.
- **Default unit**
  The default unit is used for numeric values if no other unit is provided with the parameter.

24.2 Finding the appropriate command

In the following chapters, the commands are sorted according to the menu and dialog structure of the instrument.

A list of all commands in alphabetical order ist given in the "List of Commands" at the end of this documentation.

To find the appropriate command for a setting easily, you can use the context help:

1. Enable the "Help" icon on the toolbar.

2. Tap the parameter for which you need information.

   The tooltip opens.

3. Tap the "Show Help" button in the lower right corner of the tooltip.

   The "Help" window opens and displays the comprehensive description and the corresponding remote command.

4. Tap the remote command link to open the command description.

24.3 Programming examples

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

24.3.1 Display 1260
24.3.1.1 Creating diagrams (SmartGrid) 1260
24.3.1.2 Creating zoom diagrams 1261
24.3.2 Automatic measurements 1261
24.3.2.1 Performing amplitude/time measurements 1261
24.3.2.2 Setting reference levels 1262
24.3.2.3 Waveform histograms 1263
24.3.2.4 Long-term measurements 1264
24.3.3 Mask testing 1265
24.3.3.1 Creating a user mask 1265
24.3.4 Search 1266
24.3.4.1 Searching for a pulse of specified width 1266
24.3.5 Data management 1266

24.3.5.1 Saving a screenshot to file....1266
24.3.5.2 Exporting waveform data to file....1267
24.3.5.3 Exporting measurement results to file....1273
24.3.6 Deembedding....1273
24.3.6.1 Adding cables to the measurement setup....1273
24.3.7 Protocol analysis....1273
24.3.7.1 RFFE (option R&S RTO6-K550)....1273
24.3.7.2 SENT (option R&S RTO6-K520)....1275
24.3.8 Power analysis (option R&S RTO6-K31)....1277
24.3.8.1 Auto deskew....1277
24.3.8.2 Transient response measurement....1278

24.3.1 Display

- Creating diagrams (SmartGrid)....1260
- Creating zoom diagrams....1261

24.3.1.1 Creating diagrams (SmartGrid)

The example creates 3 diagrams at different positions on the screen, and a zoom diagram. It also minimizes a wavform and adds it to another diagram

Command description in: Chapter 24.7.2.4, "Smartgrid", on page 1301

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
// Turn on display update
:SYSTem:DISPlay:UPDate ON
// Turn on Channel 2
:CHANnel2:STATe 1
// Create new diagram 'MyDiag2' below existing 'Diagram1' with waveform C2W1 in it
:LAYout:ADD 'Diagram1',VERtical,OFF,C2W1,'MyDiag2'
// Turn on Channel 3
:CHANnel3:STATe 1
// Create new diagram 'MyDiag3' right of existing 'MyDiag2'
:LAYout:ADD 'MyDiag2',HORizontal,OFF,C3W1,'MyDiag3'
// Turn on Channel 4
:CHANnel4:STATe 1
// Create new diagram 'MyDiag4' tabbed to existing 'Diagram1'
:LAYout:ADD 'Diagram1',TAB,OFF,C4W1,'MyDiag4'
// Select 'Diagram1'
:LAYout:SHOW 'Diagram1'
// Create zoom window of 'Diagram1'
:LAYout:ZOOM:ADD 'Diagram1',VERT,OFF,-10e-9,10e-9,-0.05,0.04,'MyZoom1'
```

```
// Minimize Channel 2 (Wfm1)
:LAYout:SIGNal:UNAssign C2W1
// Add Channel 2 (Wfm1) to Diagram1
:LAYout:SIGNal:ASSign 'Diagram1', C2W1

*OPC?
```

24.3.1.2 Creating zoom diagrams

The example creates a zoom diagram, sets the relative size of the zoom area, and removes the zoom diagram.

Command description in: Chapter 24.10.1, "Zoom", on page 1437.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
// Create an new zoom diagram for Diagram1
LAYout:ZOOM:ADD 'Diagram1', VERT, OFF, -10e-9, 20e-9, -0.1, 0.05, 'MyZoom1'
// Query the horizontal zoom mode, return value: ABS
LAYout:ZOOM:HORZ:MODE? 'Diagram1', 'MyZoom1'
<--ABS
// Set horizontal zoom mode to relative
LAYout:ZOOM:HORZ:MODE 'Diagram1', 'MyZoom1', REL
// Set horizontal zoom span in percent
LAYout:ZOOM:HORZ:REL:SPAN 'Diagram1', 'MyZoom1', 10
// Set horizontal zoom position in percent
LAYout:ZOOM:HORZ:REL:POS 'Diagram1', 'MyZoom1', 15
*OPC?
// Remove zoom diagram
LAYout:ZOOM:REM 'Diagram1', 'MyZoom1'
*OPC?
```

24.3.2 Automatic measurements

- Performing amplitude/time measurements 1261
- Setting reference levels 1262
- Waveform histograms 1263
- Long-term measurements 1264

24.3.2.1 Performing amplitude/time measurements

Command description is given in Chapter 24.12.1, "General settings", on page 1480 and Chapter 24.12.2, "Results", on page 1485

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

Simple frequency and amplitude measurement

```
SING;*OPC?
MEAS1:SOUR C1W1        // Configure frequency measurement
MEAS1:MAIN FREQ
MEAS1 ON
*OPC?
MEAS2:SOUR C1W1        // Configure amplitude measurement
MEAS2:MAIN AMPL
MEAS2 ON
*OPC?
MEAS1:RES:ACT?        // Get frequency result
MEAS2:RES:ACT?        // Get amplitude result
*OPC?
```

24.3.2.2 Setting reference levels

Command description in Chapter 24.12.15, "Reference levels", on page 1552

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

Manual reference level definition using relative values

Reference levels are set to 15%, 50%, and 85% of the high signal level for waveform C2W1 (= suffix 5).

```
REFLevel5:LDETection MANual
REFLevel5:LMODe REL
REFLevel5:RELative:MODE USER
REFLevel5:RELative:LOWer 15
REFLevel5:RELative:MIDDle 50
REFLevel5:RELative:UPPer 85
*OPC?
```

Manual reference level definition using absolute values

Set reference levels manually for waveform C1W1 (= suffix 2), defining high and low signal levels and the distances between signal and reference levels.

```
REFLevel2:LDETection MANual
REFLevel2:LMODe ABS
REFLevel2:ABSolute:HIGH 0.12
REFLevel2:ABSolute:TDIStance 0.03
REFLevel2:ABSolute:Low -0.12
REFLevel2:ABSolute:BDIStance 0.04
*OPC?
REFLevel2:ABSolute:MLEVel?
<-- 0
```

Set reference levels manually for waveform C1W1 (= suffix 2) by defining upper and lower reference levels and the distances between signal and reference levels.

```
REFLevel2:LDETection MANual
REFLevel2:LMODe ABS
REFLevel2:USRLevel UREF
REFLevel2:ABSolute:LLEVel 0
REFLevel2:ABSolute:BDIStance 0.02
REFLevel2:ABSolute:ULEVel 0.2
REFLevel2:ABSolute:TDIStance 0.03
*OPC?
REFLevel2:ABSolute:MLEVel?
<-- 0.1
```

Automatic level detection, peak probability

Reference levels are set to the signal levels with the highest probability values for waveform C3W1 (= suffix 8).

```
REFLevel8:LDETection Auto
REFLevel8:AUTO:MODE PPRobability
*OPC?
```

24.3.2.3 Waveform histograms

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

- Creating and reading histograms.....1263
- Exporting histogram data to file.....1264
- Transferring histogram data.....1264

Creating and reading histograms

The example creates a histogram, activates two measurements (mean and standard deviation measurements of Histogram1), and queries the results of both measurements.

Command description in:

- Chapter 24.12.1, "General settings", on page 1480
- Chapter 24.12.8.2, "Histogram measurement", on page 1527
- Chapter 24.12.2, "Results", on page 1485


```
LAY:HIST:ADD 'Histogram1', C1W1, -2.5E-007, 2.5E-007, -1.32, 5.35, OFF, VERT *OPC?

MEAS1 ON
MEAS1:HIST:SEL 'Histogram1'
MEAS1:CAT HIST
MEAS1:MAIN HME
*OPC?

MEAS2 ON
MEAS2:HIST:SEL 'Histogram1'
MEAS2:CAT HIST
MEAS2:MAIN HSTD
```

```
*OPC?

MEAS1:RES:ACT?
MEAS2:RES:ACT?
```

Exporting histogram data to file

The example writes the absolute data values of Histogram1 to `C:\Histograms\Hist1.xml` in XML format.

Command description in Chapter 24.16.5, "Waveform histogram export to file", on page 1656.

```
EXPort:HISTogram:SELect 'Histogram1'
EXPort:HISTogram:INCidence ABS
EXPort:HISTogram:NAME 'C:\Histograms\Hist1.xml'
EXPort:HISTogram:SAVE
*OPC?
```

Transferring histogram data

The example transfers the absolute values of Histogram1 to a controlling computer in ASCII format.

Command description in Chapter 24.16.5, "Waveform histogram export to file", on page 1656.

```
EXP:HIST:SEL 'Histogram1'
EXP:HIST:INC ABS
*OPC?
FORM ASC
EXP:HIST:DATA?
<--0,0,0,0,0,2037,5754804,4683496,3100169,2874565,...
```

24.3.2.4 Long-term measurements

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

- Exporting long-term measurement data to file 1264
- Transferring long-term measurement data 1265

Exporting long-term measurement data to file

The example writes the long-term data of Meas1 to `C:\Measurements\Meas1.csv` in CSV format.

Command description in Chapter 24.16.7, "Long term measurement results and measurement histogram export to file", on page 1659.

```
EXPort:MEASurement:SEL MEAS1
EXPort:MEASurement:TYPE LONGTERM
EXPort:MEASurement:NAME 'C:\Measurements\Meas1.csv'
```

```
EXPort:MEASurement:SAVE
*OPC?
```

Transferring long-term measurement data

The example transfers the long-term data of Meas1 to a controlling computer in ASCII format.

Command description in Chapter 24.16.7, "Long term measurement results and measurement histogram export to file", on page 1659.

```
MEASurement:LTM ON
MEASurement:STAT ON
EXPort:MEASurement:SELect MEAS1
EXPort:MEASurement:TYPE LONGTERM
*OPC?
FORM ASC
EXPort:MEASurement:DATA?
<--50,0.24901185771,0.24731225296,0.24703557312,0.00069270717936,0,50,....
```

24.3.3 Mask testing

- Creating a user mask 1265

24.3.3.1 Creating a user mask

Creates new user mask "MyMask" with one inner segment, and turns on the mask test.

Command description in: Chapter 24.14, "Mask testing", on page 1582.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
MTEST:ADD 'MyMask'
MTEST:SEGM:ADD 'MyMask'
MTEST:SEGM:POIN:ADD 'MyMask', 0
MTEST:SEGM:POIN:X 'MyMask', 0, 0, -20e-9
MTEST:SEGM:POIN:Y 'MyMask', 0, 0, -0.1
MTEST:SEGM:POIN:ADD 'MyMask', 0
MTEST:SEGM:POIN:X 'MyMask', 0, 1, -20e-9
MTEST:SEGM:POIN:Y 'MyMask', 0, 1, 0.1
MTEST:SEGM:POIN:ADD 'MyMask', 0
MTEST:SEGM:POIN:X 'MyMask', 0, 2, 20e-9
MTEST:SEGM:POIN:Y 'MyMask', 0, 2, 0.1
MTEST:SEGM:POIN:ADD 'MyMask', 0
MTEST:SEGM:POIN:X 'MyMask', 0, 3, 20e-9
MTEST:SEGM:POIN:Y 'MyMask', 0, 3, -0.1
MTEST:SEGM:REG 'MyMask', 0, INNER
MTEST:STAT 'MyMask', ON
*OPC?
```

24.3.4 Search

- Searching for a pulse of specified width.....1266

24.3.4.1 Searching for a pulse of specified width

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

Search for positive pulses with pulse width 12 ± 10 µs (2 µs to 22 µs).

Command description in: Chapter 24.15, "Search", on page 1603.

The usage of asynchronous commands is described in www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization".

```
SEAR:ADD 'MySearch'                  // Create a new search
SEAR:TRIG:WIDT:STAT 'MySearch',1     // Configure search type
SEAR:SOUR 'MySearch',M1              // Configure search source - here Math1
SEAR:TRIG:WIDT:RANG 'MySearch',WITH  // Configure search parameters
SEAR:TRIG:WIDT:WIDT 'MySearch',5e-4  // Configure search parameters
SEAR:TRIG:WIDT:DELT 'MySearch',5e-6  // Configure search parameters
SEAR:RES:LIM 'MySearch',1            // Set number of result lines in table to 1
*OPC?
SEAR:ALL 'MySearch';                // Initiate search for all events, asynchr. command
```

24.3.5 Data management

- Saving a screenshot to file.....1266
- Exporting waveform data to file.....1267
- Exporting measurement results to file.....1273

24.3.5.1 Saving a screenshot to file

Saves three display images in png format to the files `Print.png`, `Print_001.png`, and `Print_002.png` in the directory `C:\Temp`. To get a correct screenshot, turn on the display first.

Command description in: Chapter 24.16.8, "Screenshots", on page 1660.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
SYST:DISP:UPD ON
HCOP:DEST 'MMEM'
HCOP:DEV:LANG PNG
*OPC?
MMEM:NAME 'C:\Temp\Print.png'
HCOP:IMMediate; *OPC?
HCOP:IMM:NEXT; *OPC?
HCOP:IMM:NEXT; *OPC?
```

24.3.5.2 Exporting waveform data to file

Command description in:

- Chapter 24.16.4, "Waveform data export to file", on page 1649
- Chapter 24.16.1, "Instrument settings", on page 1640
- Chapter 24.10.4, "History", on page 1461

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

- Exporting a single waveform to XML file....1267
- Exporting raw data of a single waveform to BIN file....1267
- Exporting raw data of a measurement gate to BIN file....1268
- Exporting interleaved x/y data of a single waveform to CSV file....1269
- Exporting interleaved x/y data of a zoom to CSV file....1269
- Exporting multiple running acquisitions of a single waveform to XML file....1270
- Exporting a single acquisition of the history to BIN file....1270
- Exporting multiple acquisition of the history to XML file....1271
- Exporting and reconstructing multiple raw acquisitions of the history file....1271

Exporting a single waveform to XML file

Saves a single analog waveform completely to an XML file. Data logging is off.

The usage of asynchronous commands is described in www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization".

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
CHANnel1:WAVeform1:STATe 1
*OPC?
RUNSingle;*OPC?
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe WFM
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.xml'
EXPort:WAVeform:RAW OFF
EXPort:WAVeform:INCXvalues OFF
EXPort:WAVeform:DLOGging OFF
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
EXPort:WAVeform:SAVE
*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.xml'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.xml'
```

Exporting raw data of a single waveform to BIN file

Saves the data of a single analog waveform in integer 8-bit format (raw data) to a BIN file. Data logging is off.

Data conversion is described in "Raw (ADC direct)" on page 461.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
CHANnel1:WAVeform1:STATe 1
RUNSingle; *OPC?
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe WFM
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.bin'
EXPort:WAVeform:RAW ON
EXPort:WAVeform:INCXvalues OFF
EXPort:WAVeform:DLOGging OFF
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
EXPort:WAVeform:SAVE
*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.bin'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.bin'
```

Exporting raw data of a measurement gate to BIN file

Saves the data of a measurement gate in integer 8-bit format (raw data) to a BIN file. Data logging is off.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
CHANnel1:WAVeform1:STATe 1
MEASurement2:CATegory AMPT
MEASurement2:MAIN MEAN
MEASurement2:ENABle 1
MEASurement2:SOURce C1W1
MEASurement2:GATE:MODE ABS
MEASurement2:GATE:ABS:STARt -0.00012
MEASurement2:GATE:ABS:STOP -5e-06
MEASurement2:GATE:STATe On
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe GATE
EXPort:WAVeform:MEAS Meas2
*OPC?
RUNSingle;*OPC?
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.bin'
EXPort:WAVeform:RAW ON
EXPort:WAVeform:INCXvalues OFF
EXPort:WAVeform:DLOGging OFF
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
EXPort:WAVeform:SAVE
*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.bin'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.bin'
```

Exporting interleaved x/y data of a single waveform to CSV file

Saves the x- and y- values of a single analog waveform to a CSV file. Data logging is off.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
CHANnel1:WAVeform1:STATe 1
*OPC?
RUNSingle;*OPC?
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe WFM
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.csv'
EXPort:WAVeform:RAW OFF
EXPort:WAVeform:INCXvalues ON
EXPort:WAVeform:DLOGging OFF
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
EXPort:WAVeform:SAVE
*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.csv'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.csv'
```

Exporting interleaved x/y data of a zoom to CSV file

Saves the x- and y- values that is displayed in a zoom diagram to a CSV file. Data logging is off.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
CHANnel1:WAVeform1:STATe 1
LAYout:ZOOM:ADD 'Diagram1',HORIZONTAL,OFF,-0.00012,-5e-06,0.308,-0.092,'ExportAreaZoom'
*OPC?
RUNSingle;*OPC?
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe ZOOM
EXPort:WAVeform:ZOOM 'Diagram1', 'ExportAreaZoom'
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.csv'
EXPort:WAVeform:RAW OFF
EXPort:WAVeform:INCXvalues ON
EXPort:WAVeform:DLOGging OFF
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
EXPort:WAVeform:SAVE
*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.csv'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.csv'
```

Exporting multiple running acquisitions of a single waveform to XML file

Saves the data of 5 subsequent acquisitions of a single analog waveform to an XML file. Data logging is on.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
CHANnel1:WAVeform1:STATe 1
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe WFM
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.xml'
EXPort:WAVeform:RAW OFF
EXPort:WAVeform:INCXvalues OFF
*OPC?
ACQuire:COUNt 5
EXPort:WAVeform:DLOGging ON
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
RUNSingle;*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.xml'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.xml'
```

Exporting a single acquisition of the history to BIN file

Saves the oldest acquisition of the history to a BIN file. Data logging is off.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
CHANnel1:WAVeform1:STATe 1
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe WFM
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.bin'
EXPort:WAVeform:RAW OFF
EXPort:WAVeform:INCXvalues OFF
EXPort:WAVeform:DLOGging OFF
*OPC?
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
ACQuire:COUNt 5             //Acquire 5 waveforms
RUNSingle;*OPC?
CHANnel1:WAV1:HISTory:STATe ON
CHANnel1:WAV1:HISTory:CURRent -4;*OPC?  //Oldest waveform of 5 has index -4
EXPort:WAVeform:SAVE
*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.bin'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.bin'
```

Exporting multiple acquisition of the history to XML file

Saves the data of 5 subsequent acquisitions of the history to an XML file. Data logging is on.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
STOP;*OPC?
EXPort:WAVeform:FASTexport ON
CHANnel1:WAVeform1:STATe 1
EXPort:WAVeform:SOURce C1W1
EXPort:WAVeform:SCOPe WFM
EXPort:WAVeform:NAME 'C:\Data\DataExportWfm_analog.xml'
EXPort:WAVeform:RAW OFF
EXPort:WAVeform:INCXvalues OFF
EXPort:WAVeform:DLOGging ON
*OPC?
MMEM:DEL 'C:\Data\DataExportWfm_analog.*'
ACQuire:COUNt 5               //Acquire 5 waveforms
RUNSingle;*OPC?
CHANnel1:WAV1:HISTory:STATe ON
CHANnel1:WAV1:HISTory:STARt -4
CHANnel1:WAV1:HISTory:STOP 0
CHANnel1:WAV1:HISTory:REPLay OFF
CHANnel1:WAV1:HISTory:PLAY
*OPC?
MMEM:DATA? 'C:\Data\DataExportWfm_analog.xml'
MMEM:DATA? 'C:\Data\DataExportWfm_analog.wfm.xml'
```

Exporting and reconstructing multiple raw acquisitions of the history file

This example captures and exports 100 waveforms acquired in fast segmentation mode (minimize blind time). The data is transferred in blocks using SCPI commands.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
// --- Set data format to signed integers, 1 byte
FORMat:DATA INT,8

// --- Create history data (assuming on channel 1) ---
CHANnel1:WAVeform1:STATe 1           //turn on channel 1
ACQuire:COUNt 100                    //acquire 100 waveforms
ACQuire:SEGMented:STATe ON; *OPC?
//turn on fast segmentation, acquisition is started and data is stored in the memory


// --- Set Export variables and export data ---
EXPort:WAVeform:SOURce C1W1         //specify source for data export
EXPort:WAVeform:SCOPe WFM           //specify range: complete acq. time
MMEM:DEL "C:\Users\Public\Documents\Rohde-Schwarz\RTx\Temp\DataExportWfm_analog.*"
        //delete data from previous runs of this script
```

```
EXPort:WAVeform:NAME "C:\Users\Public\Documents\Rohde-Schwarz\RTx\Temp\DataExpWfm.bin"
        //store data in this path and filename
EXPort:WAVeform:RAW ON
//export as raw ADC integer values (saves memory) --> data needs to be converted later
EXPort:WAVeform:INCXvalues OFF
//disable time values in data file. Time can be constructed from the header file
CHANnel1:WAV1:HISTory:STATe ON      //switch to history view
EXPort:WAVeform:DLOGging ON         //enable data logging & history
EXPort:WAVeform:TIMestamps ON       //enable relative time stamp for each acq.
CHANnel1:WAV1:HISTory:STARt -99     //oldest waveform of n acq. has index =(-1)*(n-1)
CHANnel1:WAV1:HISTory:STOP 0        //newest waveforms has index 0
CHANnel1:WAV1:HISTory:REPLay OFF
CHANnel1:WAV1:HISTory:PLAY; *OPC?   //exports waveforms to defined file location.
//2 files are created: header file *.bin and waveform data file *.Wfm.bin


// --- Put files into output buffer of scope and collect ---
// The following code lines are mostly Pseudo Code.
//Sorting out the binary waveform data is more complex and require additional coding
binaryFormat = '1 byte'
//Pseudo Code, tell your language how to interpret the binary data,
//e.g. 'int8' for MATLAB
MMEM:DATA? "C:\Users\Public\Documents\Rohde-Schwarz\RTx\Temp\DataExpWfm.bin"
//Put header file into output buffer
header = readSCPIBinary(visaInstrument, binaryFormat);
//Pseudo Code, use appropriate command from your programming language,
//e.g. binblockread in MATLAB
MMEM:DATA? "C:\Users\Public\Documents\Rohde-Schwarz\RTx\Temp\DataExpWfm.Wfm.bin"
//Put data file into output buffer
//(your input buffer of the VISA resource might need to be increased)
wfmRaw = readSCPIBinary(visaInstrument, binaryFormat)
//Pseudo Code, use appropriate command from your programming language,
//e.g. binblockread in MATLAB.
//Note: Sort different acquisitions into an array separately after file transfer.


// --- Convert raw ADC values into voltage floating point values ---
// header is assumed to be a struct and the members are accessed via "." syntax.
vertOffsetByPosition = header.VerticalScale * header.VerticalPosition
conversionFactor = (1/header.NofQuantisationLevels) *
  header.VerticalScale * header.VerticalDivisionCount
for(i = 0; i<100; i++){
    wfmVolt(i) = wfmRaw(i) * conversionFactor +
      header.VerticalOffset - vertOffsetByPosition
}
// Note: Depending on settings, the waveform can contain more samples than the
//record length. Remove leading and trailing samples from the waveform.
```

24.3.5.3 Exporting measurement results to file

See:

- "Exporting histogram data to file" on page 1264
- "Transferring histogram data" on page 1264
- "Exporting long-term measurement data to file" on page 1264
- "Transferring long-term measurement data" on page 1265

24.3.6 Deembedding

24.3.6.1 Adding cables to the measurement setup

The example adds a cable to each channel and assigns an S-parameter file for each cable. Commands are explained in Chapter 24.24, "Deembedding (option R&S RTO6-K121)", on page 2558.

```
// Preset de-embedding, because *RST does not affect de-embedding
:DEEMbedding1:PRESet    // Ch1
:DEEMbedding2:PRESet    // Ch2
:DEEMbedding3:PRESet    // Ch3
:DEEMbedding4:PRESet    // Ch4

// Add cables to each channel, cable is added next to DUT at index 2 (DUT = index 1)
:DEEMbedding1:ADD 2,cabl    // Ch1
:DEEMbedding2:ADD 2,cabl    // Ch2
:DEEMbedding3:ADD 2,cabl    // Ch3
:DEEMbedding4:ADD 2,cabl    // Ch4

// Assign corresponding S2P file to channel
:DEEMbedding1:COMPonent2:SPONE 'C:\Rtx\Deembedding\RT-ZA17-1806671-0005-A.s2p'
:DEEMbedding2:COMPonent2:SPONE 'C:\Rtx\Deembedding\RT-ZA17-1806671-0005-B.s2p'
:DEEMbedding3:COMPonent2:SPONE 'C:\Rtx\Deembedding\RT-ZA17-1806671-0006-A.s2p'
:DEEMbedding4:COMPonent2:SPONE 'C:\Rtx\Deembedding\RT-ZA17-1806671-0006-B.s2p'
```

24.3.7 Protocol analysis

24.3.7.1 RFFE (option R&S RTO6-K550)

Configuring RFFE bus

Example 1: 1.8 V bus

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
//Configure source and thresholds for 1.8 V bus
STOP;*OPC?
BUS1:TYPE RFFE
BUS1:LABel "RFFE Test 2"
BUS1:RFFE:CLOCk:SOUR C2W1
BUS1:RFFE:DATA:SOUR C1W1
BUS:RFFE:PRESet V18

//Load a label list and switch on
BUS1:NEWList 'C:\Protocols\RFFE.csv'
BUS1:SYMBols ON

//Initiate single sweep
CHANnel1:STATe ON
CHANnel2:STATe ON
BUS1 ON
RUNSingle;*OPC?
```

Example 2: User-defined thresholds with threshold coupling

```
//Configure source and user-defined thresholds
STOP;*OPC?
BUS1:TYPE RFFE
BUS1:LABel "RFFE Test 1"
BUS1:RFFE:CLOCk:SOUR C2W1
BUS1:RFFE:DATA:SOUR C1W1
BUS1:RFFE:COUPling ON
BUS1:RFFE:CLOCk:THReshold:HYSTeresis 0.2
BUS1:RFFE:CLOCk:THReshold:HIGH 0.72
BUS1:RFFE:CLOCk:THReshold:LOW  0.48
```

Triggering on RFFE bus

Trigger on sequence start of sequences that have the slave address 0010, 0011, or 0100.

```
//Set trigger source to serial bus
TRIGger1:SOURce SBUS
TRIGger:MODE NORMal

//Trigger on sequence start of all sequences with slave address range
TRIGger1:RFFE:TYPE STARt
TRIG1:RFFE:SADD:CONDition INR
TRIG1:RFFE:SADD:MIN #H2  //SAD = 0010
TRIG1:RFFE:SADD:MAX #H4  //SAD = 0100
```

Searching RFFE data

Search for sequence ends of sequences if last bit of slave address is 1.

```
STOP;*OPC?
SEARch:ADD 'RFFESearch1'
```

```
SEARch:SOURce 'RFFESearch1',SBUS1
SEARch:TRIGger:RFFE:TYPE 'RFFESearch1',STOP
SEARch:TRIGger:RFFE:SADD:COND 'RFFESearch1', EQU
SEARch:TRIGger:RFFE:SADD:MIN 'RFFESearch1', 'XXX1'
RUNSingle;*OPC?              //asynchronous command
SEARch:ALL 'RFFESearch1';*OPC?
SEARch:RESult:RFFE:FCOunt? 'RFFESearchCh1'

SEARch:RESult:RFFE:SEQuence1:STATe?  'RFFESearchCh1'
<--OK    //First sequence is valid
SEARch:RESult:RFFE:SEQuence1:TYPE?  'RFFESearchCh1'
<--ERRL  //Extended Register Read Long sequence
```

24.3.7.2 SENT (option R&S RTO6-K520)

Configuring SENT bus

```
// Set protocol parameters
//*****************************************************
STOP;*OPC?
BUS:TYPE SENT
BUS:LABel "Bus 1 SENT"
BUS:SENT:DATA:SOURce C1W1
BUS:SENT:DATA:THReshold 2.0
BUS:SENT:CLKPeriod 0.000005
BUS:SENT:CLKTolerance 20.0
BUS:SENT:CRCVersion V2010
BUS:SENT:CRCMethod TLE
BUS:SENT:DNIBbles 5
BUS:SENT:PPULse NPP
BUS ON

//*****************************************************
// Load a label list and switch on
BUS:NEWList 'C:\Protocols\SENT_Labels.xml'
BUS:SYMBols ON
RUNSingle;*OPC?              //asynchronous command

//*****************************************************
// Display all results
BUS:SENT:RDSL ALL

//*****************************************************
// Display the fast channel transmission sequence
BUS:SENT:RDSL TRSQ

//*****************************************************
```

```
// Display the short serial message of slow channel
BUS:SENT:RDSL SMSG
```

Triggering on SENT bus

```
//Set trigger source to serial bus
TRIGger1:SOURce SBUS
TRIGger:MODE NORMal

//Trigger on the fast channel transmission sequence
TRIGger1:SENT:TYPE TSEQ
TRIGger1:SENT:TTYPE STDA
TRIGger1:SENT:statbit #H2
TRIGger1:SENT:DCONdition INR
BUS1:SENT:DNibbles 5
TRIGger1:SENT:DMIN #H2 //Data MIN = 0010
TRIGger1:SENT:DMAX #H4 //Data MAX = 0100
```

Searching SENT data

```
// Search for all frames with sync/calibration pulse
SEARch:TRIGger:SENT:CALibration 'Search1', ON
SEARch:ONLIne 'Search1',ON
SEARch:RESult:SORT:ASCending 'Search1', ON
SEARch:RESult:SENT:FCOunt? 'Search1'
SEARch:RESult:SENT:FRAMe1:STAT? 'Search1'
SEARch:RESult:SENT:FRAMe1:STARt? 'Search1'
SEARch:RESult:SENT:FRAMe1:STOP? 'Search1'
SEARch:RESult:SENT:FRAMe1:DATA? 'Search1'
SEARch:RESult:SENT:FRAMe1:NIBB1:VALue? 'Search1'
SEARch:RESult:SENT:FRAMe1:NIBB2:VALue? 'Search1'
SEARch:RESult:SENT:FRAMe1:CSValue? 'Search1'
SEARch:RESult:SENT:FRAMe1:SCOM? 'Search1'
SEARch:TRIGger:SENT:CALibration 'Search1', OFF

//*****************************************************
// Search Short serial message
SEARch:TRIGger:SENT:SHORt 'Search1', ON
SEARch:RESult:SENT:FCOunt? 'Search1'

SEARch:RESult:SENT:FRAMe1:STAT? 'Search1'
SEARch:RESult:SENT:FRAMe1:IDValue? 'Search1'
SEARch:RESult:SENT:FRAMe1:NIBB1:VALue? 'Search1'
SEARch:RESult:SENT:FRAMe1:NIBB2:VALue? 'Search1'
SEARch:RESult:SENT:FRAMe1:CSValue? 'Search1'

SEARch:TRIGger:SENT:SHORt 'Search1', OFF

//*****************************************************
// Search Enhanced serial message
SEARch:TRIGger:SENT:ENHanced 'Search1', ON
```

```
SEARch:RESult:SENT:FCOunt? 'Search1'

SEARch:RESult:SENT:FRAMe1:STAT? 'Search1'
SEARch:RESult:SENT:FRAMe1:IDValue? 'Search1'
SEARch:RESult:SENT:FRAMe1:NIBB1:VALue? 'Search1'
SEARch:RESult:SENT:FRAMe1:NIBB2:VALue? 'Search1'
SEARch:RESult:SENT:FRAMe1:CSValue? 'Search1'

SEARch:TRIGger:SENT:ENHanced 'Search1', OFF
```

24.3.8 Power analysis (option R&S RTO6-K31)

24.3.8.1 Auto deskew

Configures the voltage and current probes for power measurements and executes the auto deskew.

Command description in Chapter 24.23.1, "General", on page 2520 and Chapter 24.23.2, "Deskew", on page 2520.

If the instrument refuses to accept `POWer` commands, activate the power mode using `:POWer:ENABle`.

In the following example `*OPC` prevents overlapping execution of asynchronous commands.

```
*RST; *OPC?
STOP;*OPC?
//Activate two channels
:CHANnel1:STATe 1
:CHANnel2:STATe 1

//Activate power
:POWer:ENABle

//Select current and voltage sources
:POWer:SOURce:VOLTage1 CHANnel1
:POWer:SOURce:CURRent1 CHANnel2
*OPC?

//Configure voltage probe on CH1 manually
//Not necessary if you use an active R&S voltage probe that is recognized by the instrument
:PROBe1:SETup:ATTenuation:MODE Manual
:PROBe:SETup:ATTenuation:DEFProbe ZD01a100
*OPC?
//selected high voltage differential probe 1:100
```

```
//Configure current probe on CH2 manually
//Always required because R&S current probes are not recognized automatically
PROBe2:SETup:ATTenuation:MODE Manual
PROBe2:SETup:ATTenuation:DEFProbe ZC20
*OPC?
//select 20MHz current probe ZC10 also possible

//Start deskew
//Overwrites the skew offset of CH2 (current probe), because :POWer:DESKew:RESet? == 1
//writes a user-defined preset file (UserDefinedPreset_AutoDeskew.dfl) and
//activates the user defined preset, because :POWer:DESKew:UDPReset? == 1
:PoWer:DESKew:EXECute;

//Check result
CHANnel2:SKEW:MAN?
CHANnel2:SKEW:TIMe?
POWer:DESKew:TIMe?
*OPC?
```

Effect of *RST and loading user-defined preset

Note that `*RST` resets the deskew values.

You can reload the deskew values as follows:

```
*RST;*OPC?
STOP;*OPC?
:POWer:ENABle
//Select voltage and current sources
:POWer:SOURce:VOLTage1 CHANnel1
:POWer:SOURce:CURRent1 CHANnel2
*OPC?
//Reload deskew values
:POWer:DESKew:CURRent
*OPC?
//Load default saveset after FW restart
MMEM:RCL 'C:\Users\Public\Documents\Rohde-Schwarz\RTx\SaveSets
\UserDefinedPreset_AutoDeskew.dfl'
```

24.3.8.2 Transient response measurement

Configures and executes a transient response measurement.

Command description in Chapter 24.23.15, "Transient response", on page 2554.

Make sure to configure and deskew the probes before the measurement, see Chapter 24.3.8.1, "Auto deskew", on page 1277.

```
//Activate power
*RST; *OPC?
```

```
:POWer:ENABle

//Expected smps frequency
:POW:TRANsient:FREQ         12500000

:POWer:TRANsient:AUToscale AUTO

:POWer:TRANsient:SIGHigh 0.1
:POWer:TRANsient:SIGLow    0.025
:POWer:TRANsient:HYSTeresis 20


//Trigger channel
:POWer:TRANsient:TRGC CHAN2

//Edge trigger slope
:POWer:TRANsient:TRGS POS

//Trigger level
:POWer:TRANsient:TRGL 0.08

//Run measurement
:POWer:TRANsient:EXECute;*OPC?

//Query results
:POWer:TRANsient:RESult? SETTling
:POWer:TRANsient:RESult? PEAKtime
:POWer:TRANsient:RESult? DELay
:POWer:TRANsient:RESult? RTIMe
:POWer:TRANsient:RESult? OVERshoot

//Add to report
:Power:TRANsient:REPort:Add
```

24.4 Frequently used parameters and suffixes

This chapter describes in general those parameters and suffixes that are used in several subsystems.

24.4.1 Waveform suffix

The numeric waveform suffix is used in some commands, for example, to indicate the source waveform number from which the reference level is taken, and to assign color tables to waveforms.

Depending on the command, not all suffix values are supported. For example, in REFLevel commands, only suffixes 2 to 21 are allowed. The range of supported suffix numbers is indicated in the description of the individual commands.

NOTICE

Suffix 1

Suffix 1 means that no waveform is assigned. The first waveform C1W1 corresponds to suffix number 2.

Waveform number	Description
1	None
2	C1W1 (channel 1, waveform 1)
3	C1W2 (channel 1, waveform 2)
4	C1W3 (channel 1, waveform 3)
5	C2W1 (channel 2, waveform 1)
6	C2W2 (channel 2, waveform 2)
7	C2W3 (channel 2, waveform 3)
8	C3W1 (channel 3, waveform 1)
9	C3W2 (channel 3, waveform 2)
10	C3W3 (channel 3, waveform 3)
11	C4W1 (channel 4, waveform 1)
12	C4W2 (channel 4, waveform 2)
13	C4W3 (channel 4, waveform 3)
14 to 17	Math waveforms: M1, M2, M3, M4 M5 to M8: suffixes 119 to 122
18 to 21	Reference waveforms: R1, R2, R3, R4
22 to 25	XY-waveforms: XY1, XY2, XY3, XY4
26 to 33	Measurement results: MRESult1, MRESult2, MRESult3, MRESult4, MRESult5, MRESult6, MRESult7, MRESult8 MRESult1 are the results of measurement group 1 and so on.
34 to 35	not used
36 to 39	Serial buses: SBUS1, SBUS2, SBUS3, SBUS4
40 to 55	Digital channels: D0 to D15 (option R&S RTO6-B1)
56 to 59	Digital buses: MSO1, MSO2, MSO3, MSO4 (option R&S RTO6-B1)
60	not used
61 to 68	Track waveforms: TRK1, TRK2, TRK3, TRK4, TRK5, TRK6, TRK7, TRK8

Waveform number	Description
69 to 71	not used
72 to 75	Spectrograms: SG1, SG2, SG3, SG4. Available for option R&S RTO6-K37. SG5 to SG8: suffixes 119 to 122 . Available for option R&S RTO6-K133.
76 to 83	Timeline spectrums: SG1TL1, SG1TL2, SG2TL1, SG2TL2, SG3TL1, SG3TL2, SG4TL1, SG4TL2. Available for option R&S RTO6-K37 TL1 is timeline 1, TL2 is timeline 2 SG5TL1 to SG8TL2: suffixes 123 to 130. Available for option R&S RTO6-K133.
84 to 87	Eye diagrams: EYE1, EYE2, EYE3, EYE4 Available for eye diagram analysis with option R&S RTO6-K91
88 to 91	Voltage input channels of multi-channel probe R&S RT-ZVC (probe 1) R&S RT-ZVC04: Z1V1 | Z1V2 | Z1V3 | Z1V4 R&S RT-ZVC02: Z1V1 | Z1V2. Suffixes 90 to 91 are not available.
92 to 95	Current input channels of multi-channel probe R&S RT-ZVC (probe 1) R&S RT-ZVC04: Z1I1 | Z1I2 | Z1I3 | Z1I4 R&S RT-ZVC02: Z1I1 | Z1I2. Suffixes 94 to 95 are not available.
96 to 99	Voltage input channels of multi-channel probe R&S RT-ZVC (probe 2) R&S RT-ZVC04: Z2V1 | Z2V2 | Z2V3 | Z2V4 R&S RT-ZVC02: Z2V1 | Z2V2. Suffixes 98 to 99 are not available.
100 to 103	Current input channels of multi-channel probe R&S RT-ZVC (probe 2) R&S RT-ZVC04: Z2I1 | Z2I2 | Z2I3 | Z2I4 R&S RT-ZVC02: Z2I1 | Z2I2. Suffixes 102 to 103 are not available.
104 to 107	not used
108 to 111	Resulting waveforms of differential deembedding: DIFF1 | DIFF2 | COMMON1 | COMMON2
112 to 114	Resulting waveforms of TDR/TDT analysis (option R&S RTO6-K130): TDRZ | TDRH | TDRR
115 to 118	Math waveforms: M5, M6, M7, M8 M1 to M4: suffixes 14 to 17
119 to 122	Spectrograms: SG5, SG6, SG7, SG8. Available for option R&S RTO6-K133. SG1 to SG4: suffixes 72 to 75. Available for option R&S RTO6-K37.
123 to 130	Timeline spectrums: SG5TL1, SG5TL2, SG6TL1, SG6TL2, SG7TL1, SG7TL2, SG8TL1, SG8TL2. Available for option R&S RTO6-K133. TL1 is timeline 1, TL2 is timeline 2 SG1TL1 to SG4TL2: suffixes 76 to 83. Available for option R&S RTO6-K37.
131 to 140	HP1 | HP2 | HP3 | HP4 | HP5 | HP6 | HP7 | HP8 | HP9 | HP10 Available for option R&S RTO6-K133.
141	not used
142 to 144	BATHtub1 | BMEas1 | SRESponse1 Bathtub diagram, bathtub measurement, and step response of 1st advanced jitter measurement, option R&S RTO6-K133/K134.

Waveform number	Description
145 to 151	SRHist1 | SDHist1 | TJHist1 | DJHist1 | DDJHist1 | PJHist1 | RJOHist1 Histograms of 1st advanced jitter measurement, option R&S RTO6-K133/K134. SR: symbol rate, SD: symbol duration, TJ: total jitter, DDJ: data-dependent jitter, PJ: periodic jitter, RJO: random jitter.
152, 154- 158	SRTRack1 | TJTRack1 | DJTRack1 | DDJTrack1 | PJTRack1 | RJOTRack1 Tracks of 1st advanced jitter measurement, option R&S RTO6-K133/K134.
153	not used
159 to 163	TJSPectrum1 | DDJSpectrum1 | PJOSPectrum1 | RJOSPectrum1 | SEYE1 Spectrograms and synthetic eye diagram of 1st advanced jitter measurement, option R&S RTO6-K133/K134.
164	not used
165 to 167	BATHtub2 | BMEas2 | SRESponse2 Bathtub diagram, bathtub measurement, and step response of 2nd advanced jitter measurement, option R&S RTO6-K133/K134.
168 to 174	SRHist2 | SDHist2 | TJHist2 | DJHist2 | DDJHist2 | PJOHist2 | RJOHist2 Histograms of 2nd advanced jitter measurement, option R&S RTO6-K133/K134.
175-177, 179-181	SRTRack2 | SDTRack2 | TJTRack2 | DDJTrack2 | PJOTrack2 | RJOTrack2 Tracks of 1st advanced jitter measurement, option R&S RTO6-K133/K134.
178	not used
182 to 186	TJSPectrum2 | DDJSpectrum2 | PJOSPectrum2 | RJOSPectrum2 | SEYE2 Spectrograms and synthetic eye diagram of 2nd advanced jitter measurement, option R&S RTO6-K133/K134.
187 to 188	NBAThtub1 | NBMeas1 Noise bathtub diagram and noise bathtub measurement of 1st noise measurement, option R&S RTO6-K134.
189 to 193	TNHist1 | DNHist1 | DDNHist1 | PNHist1 | RNOHist1 | Histograms of 1st noise measurement, option R&S RTO6-K134. TN: total noise, DN: deterministic noise, DDN: data-dependent noise, PJ: periodic noise, RNO: random noise.
194, 196 to 198	TNTRack1 | DDNTrack1 | PNTRack1 | RNOTRack1 Tracks of 1st noise measurement, option R&S RTO6-K134.
195	not used
199 to 202	TNSPectrum1 | DDNSpectrum1 | PNSPectrum1 | RNOSPectrum1 Spectrograms of 1st noise measurement, option R&S RTO6-K134.
203 to 204	NBAThtub2 | NBMeas2 Noise bathtub diagram and noise bathtub measurement of 2nd noise measurement, option R&S RTO6-K134.
205 to 209	TNHist2 | DNHist2 | DDNHist2 | PNHist2 | RNOHist2 Histograms of 2nd noise measurement, option R&S RTO6-K134.

Waveform number	Description
210, 212 to 214	TNTRack2 | DDNTrack2 | PNTRack2 | RNOTRack2 Tracks of 2nd noise measurement, option R&S RTO6-K134.
211	not used
215 to 218	TNSPectrum2 | DDNSpectrum2 | PNSPectrum2 | RNOSpectrum2 Spectrograms eye diagram of 2nd noise measurement, option R&S RTO6-K134.
219 to 221	PJDHist1 | BUJHist1 | RJHist1 Histograms of 1st jitter measurement, option R&S RTO6-K134. BUJ: (Other) bounded uncorrelated jitter.
222	PJDTrack1 Tracks of 1st data dependent and periodic jitter noise measurement, option R&S RTO6-K134.
223	PJDSpectrum1 Spectrograms of 1st data dependent and periodic jitter noise measurement, option R&S RTO6-K134.
224 to 226	PJDHist2 | BUJHist2 | RJHist2 Histograms of 2nd noise measurement, option R&S RTO6-K134.
227	PJDTrack2 Tracks of 2nd data dependent and periodic jitter noise measurement, option R&S RTO6-K134.
228	PJDSpectrum2 Spectrograms of 2nd data dependent and periodic jitter measurement, option R&S RTO6-K134.
229 to 231, 234 to 236	PNDHist1 | BUNHist1 | RNHist1 | PNDHist2 | BUNHist2 | RNHist2 Histograms of noise measurement, option R&S RTO6-K134. BUN: (Other) bounded uncorrelated noise.
232	PNDTrack1 Tracks of 1st data dependent and periodic noise measurement, option R&S RTO6-K134.
233	PNDSpectrum1 Spectrograms of 1st data dependent and periodic noise measurement, option R&S RTO6-K134.
237	PNDTrack2 Tracks of 2nd data dependent and periodic noise measurement, option R&S RTO6-K134.
238	PNDSpectrum2 Spectrograms of 2nd data dependent and periodic noise measurement, option R&S RTO6-K134.

24.4.2 Waveform parameter

Many commands requires one of the waveforms to be specified as source. The following table lists all waveforms. For each command using a waveform parameter, the available waveforms are specified in the command description.

Waveform	Description
C1W1 | C1W2 | C1W3	Channel 1 waveforms
C2W1 | C2W2 | C2W3	Channel 2 waveforms
C3W1 | C3W2 | C3W3	Channel 3 waveforms
C4W1 | C4W2 | C4W3	Channel 4 waveforms
M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8	Math waveforms
R1 | R2 | R3 | R4	Reference waveforms
XY1 | XY2 | XY3 | XY4	XY-waveforms
MRESult1 | MRESult2 | MRESult3 | MRESult4 | MRESult5 | MRESult6 | MRESult7 | MRESult8	Measurement results
SBUS1 | SBUS2 | SBUS3 | SBUS4	Serial buses
D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15	Digital channels (option R&S RTO6-B1)
MSOB1 | MSOB2 | MSOB3 | MSOB4	Digital buses (option R&S RTO6-B1
TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8	Track waveforms
CDRSW1 = CDRSw1, CDRSW2 = CDRSw2 CDRHW = CDRHw	Generated clock signals
SG1 | SG2 | SG3 | SG4	Spectrograms (option R&S RTO6-K37)
SG5 | SG6 | SG7 | SG8	Advanced jitter analysis (option R&S RTO6-K133)
SG1TL1, SG1TL2, SG2TL1, SG2TL2, SG3TL1, SG3TL2, SG4TL1, SG4TL2	Spectrograms (option R&S RTO6-K37)
SG5TL1, SG5TL2, SG6TL1, SG6TL2, SG7TL1, SG7TL2, SG8TL1, SG8TL2	Advanced jitter analysis (option R&S RTO6-K133)
EYE1 | EYE2 | EYE3 | EYE4	Available for eye diagram analysis with option R&S RTO6-K91
Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 Z1V1 | Z1V2 | Z1I1 | Z1I2 | Z2V1 | Z2V2 | Z2I1 | Z2I2	Input channels of multi-channel probe R&S RT-ZVC04 Input channels of multi-channel probe R&S RT-ZVC02
DIFF1 | DIFF2 | COMMON1 | COMMON2	Resulting waveforms of differential deembedding
TDRZ | TDRH | TDRR	Resulting waveforms of TDR/TDT analysis (option R&S RTO6-K130)
SEYE1 | SEYE2	Available for synthetic eye diagram analysis with option R&S RTO6-

Waveform	Description
TJSPectrum1 | TJSPectrum2 | DDJSpectrum1 | DDJSpectrum2 | PJSPectrum1 | PJSPectrum2 | RJOSPectrum | RJOSPectrum1	Spectrograms of the advanced jitter analysis components (option R&S RTO6-K133/K134) TJ: total jitter DDJ: data-dependent jitter PJ: periodic jitter RJO: random jitter
BATHtub1 | BATHtub2	Bathtub diagrams of the advanced jitter analysis components (option R&S RTO6-K133/K134)
BMEas1 | BMEas2	Bathtub measurements of the advanced jitter analysis components (option R&S RTO6-K133/K134)
SRTRack1 | SRTRack2 | SDTRack1 | SDTRack2 | TJTRack1 | TJTRack2 | DJTRack1 | DJTRack2 | DDJTrack1 | DDJTrack2| PJOTRack1 | PJOTRack2 | RJOTRack1| RJOTRack2	Tracks of the advanced jitter analysis components (option R&S RTO6-K133/K134) SR: Symbol rate SD: Symbol duration
SRESponse1 | SRESponse2	Step response of the advanced jitter analysis (option R&S RTO6-K133/K134)
TNSPectrum1 | TNSPectrum2 | DDNSpectrum1 | DDNSpectrum2 | PNSPectrum1 | PNSPectrum2 | RNOSpectrum1 | RNOSpectrum2	Spectrograms of the advanced jitter analysis components (option R&S RTO6-K134) TN: total noise DDN: data-dependent noise PN: periodic noise RNO: random noise
NBAThtub1 | NBAThtub2	Bathtub diagrams of the noise analysis components (option R&S RTO6-K134)
NBMeas1 | NBMeas2	Bathtub measurements of the noise analysis components (option R&S RTO6-K134)
TNTRack1 | TNTRack2 | DNTRack1 | | DNTRack2 | DDNTrack1 | DDNTrack2 | PNTRack1 | PNTRack2 | RNOTRack1 | RNOTRack2	Tracks of the noise analysis components (option R&S RTO6-K134)

24.4.3 Slope parameter

The slope parameter is used with several trigger and search condition commands.

Slope	Description
POSitive	Rising edge, that is a positive voltage change.
NEGative	Falling edge, that is a negative voltage change
EITHer	rising as well as the falling edge.

24.4.4 Polarity parameter

The polarity parameter is used with several trigger and search condition commands.

Polarity	Description
POSitive	Positive going pulses.
NEGative	Negative going pulses.
EITHer	Both positive and negative going pulses.

24.4.5 Event parameter

The event parameter is used with commands defining an action for mask testing, limit checks and margin checks.

Event	Description
NOACtion	The action is not initiated.
SUCCess	The action is initiated if the operation finished successfully: • Limits or margins were not exceeded during the entire measurement • Mask test passed
VIOLation	The action is initiated if the operation finished with error: • Limits or margins were violated during the measurement • Mask test failed

24.4.6 Bit pattern parameter

Bit pattern parameter are required with commands triggering on address, identifier, or data pattern.

To set the pattern value, you can use either a numeric parameter as defined in the SCPI standard, or a string parameter.

Bit pattern in numeric parameter

In a numeric parameter, the values are listed byte-by-byte, with bytes separated by commas and MSB first. The default numeral format is decimal, other formats can be indicated by a format identifier (#B = binary, #H = hexadecimal, #Q = octal). Currently, no format for signed values is available.

Example: Parameter with three bytes, decimal byte values are 10, 20, 30. The examples are given for CAN, the bit pattern in other commands is defined in the same way.

- `TRIGger:CAN:DMIN 10,20,30`
- `TRIGger:CAN:DMIN #B00001010,#B00010100,#B00011110`
- `TRIGger:CAN:DMIN #H0A,#H14,#H1E`
- `TRIGger:CAN:DMIN #Q012,#Q024,#Q036`

Bit pattern in string parameter

In a string, the complete binary pattern is written without separation of bytes, for example:

```
TRIGger:CAN:DMIN '000010100001010000011110'
```

Unlike a numeric parameter, the string parameter accepts wildcards for single bits (X = don't care). Whether wildcards can be used or not depends on the remote command. Usually, address and identifier parameter require unique patterns while data parameters may contain wildcards.

Mostly the length of the bit pattern is defined, for example, by the I²C address type, the CAN identifier type, or the data length code. In theses cases, it is recommended that you enter the complete bit pattern. If you enter a shorter pattern, the instrument fills up the pattern with X bits to the right of the defined pattern.

Example: You want to trigger on an 11 bit CAN address and enter the bit pattern '11100011' (8 bits only). The instrument uses the pattern '11100011XXX' for triggering.

Query for a pattern

The pattern format for the return value of a pattern is defined by the `FORMat:BPATtern` command.

24.5 Common commands

Common commands are described in the IEEE 488.2 (IEC 625-2) standard. These commands have the same effect and are employed in the same way on different devices. The headers of these commands consist of "*" followed by three letters. Many common commands are related to the Status Reporting System.

Available common commands:

*CAL?....1288
*CLS....1288
*ESE....1288
*ESR?....1288
*IDN?....1288
*IST?....1289
*OPC....1289
*OPT?....1289
*PCB....1289
*PRE....1290
*PSC....1290
*RCL....1290
*RST....1290
*SAV....1290
*SRE....1291
*STB?....1291
*TRG....1291
*TST?....1291
*WAI....1292

***CAL?**

Starts a self-alignment of the instrument, and then queries a status response. Return values ≠ 0 indicate an error.

Return values:

<State> 0: no error
1: alignment failed
2: not aligned, e.g. init
3: device needs longer warmup time before selfalignment can start
4: input signal connected during selfalignment

Usage: Query only

***CLS**

Clear status

Sets the status byte (STB), the standard event register (ESR) and the `EVENt` part of the `QUEStionable` and the `OPERation` registers to zero. The command does not alter the mask and transition parts of the registers. It clears the output buffer.

Usage: Setting only

***ESE** <Value>

Event status enable

Sets the event status enable register to the specified value. The query returns the contents of the event status enable register in decimal form.

Parameters:

<Value> Range: 0 to 255

***ESR?**

Event status read

Returns the contents of the event status register in decimal form and then sets the register to zero.

Return values:

<Contents> Range: 0 to 255

Usage: Query only

***IDN?**

Identification

Returns the instrument identification.

Return values:

<ID> "Rohde&Schwarz,<device type>,<part number>/<serial number>,<firmware version>"

Example: `Rohde&Schwarz,RTO6,1802.0001k04/200153,5.0.0.2`
Model assignment:

Usage: Query only

***IST?**

Individual status query

Returns the contents of the IST flag in decimal form. The IST flag is the status bit which is sent during a parallel poll.

Return values:

<ISTflag> 0 | 1

Usage: Query only

***OPC**

Operation complete

Sets bit 0 in the event status register when all preceding commands have been executed. This bit can be used to initiate a service request. The query writes a "1" into the output buffer when all preceding commands have been executed, which is useful for command synchronization.

***OPT?**

Option identification query

Queries the options included in the instrument. For a list of all available options and their description, refer to the data sheet.

Return values:

<Options> The query returns a list of options. The options are returned at fixed positions in a comma-separated string. A zero is returned for options that are not installed.

Usage: Query only

***PCB** <Address>

Pass control back

Indicates the controller address to which remote control is returned after termination of the triggered action.

Setting parameters:

<Address> Range: 0 to 30

Usage: Setting only

***PRE** <Value>

Parallel poll register enable

Sets parallel poll enable register to the indicated value. The query returns the contents of the parallel poll enable register in decimal form.

Parameters:
<Value> Range: 0 to 255

***PSC** <Action>

Power on status clear

Determines whether the contents of the `ENABle` registers are preserved or reset when the instrument is switched on. Thus a service request can be triggered when the instrument is switched on, if the status registers ESE and SRE are suitably configured. The query reads out the contents of the "power-on-status-clear" flag.

Parameters:
<Action> 0 | 1

0
The contents of the status registers are preserved.

1
Resets the status registers.

***RCL** <Number>

Recall

Loads the instrument settings from an intermediate memory identified by the specified number. The instrument settings can be stored to this memory using the command `*SAV` with the associated number.

***RST**

Reset

Sets the instrument to a defined default status. The default settings are indicated in the description of commands.

Usage: Setting only

***SAV** <Number>

Save

Stores the current instrument settings under the specified number in an intermediate memory. The settings can be recalled using the command `*RCL` with the associated number.

***SRE** <Contents>

Service request enable

Sets the service request enable register to the indicated value. This command determines under which conditions a service request is triggered.

Parameters:

<Contents> Contents of the service request enable register in decimal form. Bit 6 (MSS mask bit) is always 0.

Range: 0 to 255

***STB?**

Status byte query

Reads the contents of the status byte in decimal form.

Usage: Query only

***TRG**

Trigger

Triggers all actions waiting for a trigger event. In particular, *TRG generates a manual trigger signal. This common command complements the commands of the `TRIGger` subsystem.

Usage: Event

***TST?**

Self-test query

Initiates self-tests of the instrument and returns an error code.

Return values:

<ErrorCode> **integer > 0 (in decimal format)**
An error occurred.
(For details, see the Service Manual supplied with the instrument).

0
No errors occurred.

Usage: Query only

***WAI**

Wait to continue

Prevents servicing of the subsequent commands until all preceding commands have been executed and all signals have settled (see also command synchronization and `*OPC`).

Usage: Event

24.6 General remote settings

This chapter describes commands that have effect on many other remote commands in different applications of the instrument.

FORMat[:DATA]..... 1292
FORMat:BORDer..... 1293
FORMat:BPATtern..... 1294
SYSTem:DISPlay:UPDate..... 1294
SYSTem:KLOCk..... 1294
GPIB:ADDRess..... 1294
GPIB:TERMinator..... 1295
SYSTem:DISPlay:MESSage:STATe..... 1295
SYSTem:DISPlay:MESSage[:TEXT]..... 1295
SYSTem:LANGuage..... 1295

FORMat[:DATA] <Format>, [<Length>]

Selects the data type that is used for transmission of data from analog channels, math and reference waveforms, and some measurement results from the instrument to the controlling computer.

The command sets the data format for the following query commands:

- `CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?`
- `CALCulate:MATH<m>:DATA[:VALues]?`
- `REFCurve<m>:DATA[:VALues]?`
- `EXPort:HISTogram:DATA?` on page 1657
- `EXPort:MEASurement:DATA?` on page 1660
- `MEASurement<m>:TRACk:DATA[:VALues]?` on page 1546

Parameters:

<Format>,[<Length>] ASCii | REAL,32 | INT,8 | INT,16

ASCii
Data values are returned in ASCII format as a list of comma-separated values in floating point format. The length can be omitted. It is 0 which means that the instrument selects the number of digits to be returned. The query returns both values (`ASC,0`).

REAL,32
The data is stored as binary data (Definite Length Block Data according to IEEE 488.2). Each waveform value is formatted in 32-Bit IEEE 754 Floating Point Format.
The schema of the result string is as follows:
`#41024<value1><value2>…<value n>` with:
`#4` = number of digits (= 4 in the example) of the following number
`1024` = number of following data bytes (= 1024 in the example)
`<value>` = 4-byte floating point values
For large data (≥ 1 GB), the result string starts with "#(data length)". The number inside the parentheses indicates the real data length in bytes.

INT,8 | INT,16
Signed integer data with length 8 bit or 16 bit. It defines that `CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?` returns the raw sample data of the ADC as integers. If format of the waveform data differs from the defined export format, the instrument converts the data to the required length.
The result string has the same schema as the REAL format.
Fot INT,16 you can set the byte order using the `FORMat:BORDer` command.
Data conversion is described in "Raw (ADC direct)" on page 461.
For digital channel data, math and histogram data, INT formats are not available.
`EXPort:WAVeform:INCXvalues` must be set `OFF`.

*RST: ASCii

Example:
```
FORMat:DATA REAL,32
FORMat:DATA?
REAL,32
```

Usage: SCPI confirmed

FORMat:BORDer <ByteOrder>

Sets the endianess.

The command is only relevant for raw data export in high definition mode (16 bit word length).

Parameters:
<ByteOrder> LSBFirst | MSBFirst

LSB first: little endian, least significant byte first
MSB first: big endian, most significant byte first

*RST: LSBFirst

Usage: Asynchronous command

FORMat:BPATtern <BitPatternFormat>

Sets the number format for remote bit pattern queries on serial protocols.

Parameters:

<BitPatternFormat> DEC | HEX | OCT | BIN | ASCII | ASCii | STRG
ASCII = ASCii
*RST: HEX

Usage: Asynchronous command

SYSTem:DISPlay:UPDate <Enable>

Defines whether the display is updated while the instrument is in the remote state. If the display is switched off, the normal GUI is replaced by a static image while the instrument is in the remote state. Switching off the display can speed up the measurement. This is the recommended state.

See also: Chapter 22.5.3.2, "Using the display during remote control", on page 1245

Parameters:

<Enable> **ON | 1:** Display is shown and updated during remote control
OFF | 0: Display shows static image during remote control

Example: `SYSTem:DISPlay:UPDate 1`
Switch on the display update.

SYSTem:KLOCk <Enable>

Locks or unlocks the local controls of the instrument. This includes the front panel keys, the keyboard, or other local interfaces. except for the "View" button on the display.

Parameters:

<Enable> **ON | 1:** Locks the local keys
OFF | 0: Keys are unlocked

Usage: SCPI confirmed

GPIB:ADDRess <Address>

Sets the GPIB address of the instrument. Changing the address has major effects on the communication to the remote computer.

Parameters:

<Address> Range: 0 to 30
Increment: 1
*RST: 20

Usage: Asynchronous command

GPIB:TERMinator <Terminator>

Specifies the symbol that is used as a terminator in GPIB communication.

Parameters:

<Terminator> LFEoi | EOI

*RST: EOI

Usage: Asynchronous command

SYSTem:DISPlay:MESSage:STATe <DispMessState>

Enables and disables the display of an additonal text in remote control.

To define the text, use `SYSTem:DISPlay:MESSage[:TEXT]`.

Parameters:

<DispMessState> ON | OFF

*RST: OFF

Usage: Asynchronous command

Firmware/software: Version 2.70

SYSTem:DISPlay:MESSage[:TEXT] <DisplayMessage>

Defines an additional text that is displayed during remote control operation.

To enable the text display, use `SYSTem:DISPlay:MESSage:STATe` on page 1295.

Parameters:

<DisplayMessage> String that contains the text.

Usage: Asynchronous command

Firmware/software: Version 2.70

SYSTem:LANGuage <Language>

Defines the remote control behavior of the instrument and sets the remote control command set.

Parameters:

<Language> String value. Available values:
'SCPI': R&S RTO6 remote command set is used.
"DPO7000" or 'TDS540': Compatible remote command set of Tektronix oscilloscopes DPO7000 or TDS540 is used.
If one of these emulation modes is used, you can define alternative responses to the `IDN*?` and `OPT*?` commands on the "Menu" > "Settings" > "System" > "Remote" > "SCPI Emulation" tab.

Usage: Asynchronous command

Options: R&S RTO6--K301

24.7 Instrument setup

24.7.1 System

SYSTem:PRESet 1296
SYSTem:DATE 1296
SYSTem:TIME? 1296
SYSTem:DEVice:ID? 1297
DIAGnostic:SERVice:COMPutername 1297

SYSTem:PRESet

Resets the instrument to the factory default settings, to the initial state. Factory settings comprise all instrument settings, including display, intensity and transparency settings. After loading factory defaults, perform a self-alignment to synchronize the signal data.

Usage: Event

SYSTem:DATE <Year>, <Month>, <Day>

Sets the date of the internal calendar.

Parameters:

<Year> Year, to be entered as a four-digit number (including the century and millennium information)

<Month> Month, 1 (January) to 12 (December)

<Day> Day, 1 to the maximum number of days in the specified month

*RST: does not affect the date settings

Example: `SYSTem:DATE?`
Returned value: `2011,09,13`

Usage: SCPI confirmed

SYSTem:TIME? <Hour>, <Minute>, <Second>

Returns the UTC (Universal Time Coordinated) of the internal clock. To define the current local time, use the time zone setting of the operating system

Example: `SYSTem:TIME?`
Returned value: `15,09,20`. UTC is 15:09:20.

Usage: Query only
SCPI confirmed

SYSTem:DEVice:ID?

Returns the instrument ID - that is the part number and the serial number.

Return values:

<ID> String containing the part number and the serial number

Example: `1316.1000K24-001122-jT`

Usage: Query only

DIAGnostic:SERVice:COMPutername <ComputerName>

The query returns the computer name that is currently defined. The computer name is required when configuring a network.

The setting command changes the computer name. The change takes effect after the next reboot of the computer.

Parameters:

<ComputerName> Name string

Usage: Asynchronous command

24.7.2 Appearance

24.7.2.1 Colors appearance

DISPlay:COLor:SIGNal<m>:COLor....1297
DISPlay:COLor:SIGNal<m>:ASSign....1297
DISPlay:COLor:SIGNal<m>:USE....1298

DISPlay:COLor:SIGNal<m>:COLor <Value>

Suffix:

<m> 1...218
Waveform number, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:

<Value> Decimal value of the ARGB color. Use the color dialog box on the instrument to get the hex value of the color, and convert the hex value to a decimal value.
0 is fully transparent black.
4278190080 (dec) = FF000000 (hex) is opaque black.
4294967295 (dec) = FFFFFFFF (hex) is opaque white.

DISPlay:COLor:SIGNal<m>:ASSign <ColorTable>

Assigns the color table to the specified signal.

Suffix:
<m> 1...218
Waveform number, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<ColorTable> Color table name to be assigned to the signal.

DISPlay:COLor:SIGNal<m>:USE <State>

If enabled, the selected waveform is displayed according to its assigned color table.

If this option is disabled, the default color table is used, i.e. the intensity of the specific signal color varies according to the cumulative occurance of the values.

Suffix:
<m> 1...218
Waveform number, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<State> ON | OFF

24.7.2.2 Color tables

DISPlay:COLor:PALette:ADD 1298
DISPlay:COLor:PALette:REMove 1298
DISPlay:COLor:PALette:COUNt? 1299
DISPlay:COLor:PALette:POINt:ADD 1299
DISPlay:COLor:PALette:POINt:INSert 1299
DISPlay:COLor:PALette:POINt:REMove 1299
DISPlay:COLor:PALette:POINt[:VALue] 1299
DISPlay:COLor:PALette:POINt:COUNt? 1300

DISPlay:COLor:PALette:ADD <Name>

Adds a new color table with the specified name.

Setting parameters:
<Name> color table

Usage: Setting only

DISPlay:COLor:PALette:REMove <Name>

Removes the specified color table.

Setting parameters:
<Name> color table

Usage: Setting only

DISPlay:COLor:PALette:COUNt?

Queries the number of configured color maps.

Usage: Query only

DISPlay:COLor:PALette:POINt:ADD <PaletteName>

Appends a new row at the end of the color table.

Setting parameters:
<PaletteName> color table

Usage: Setting only

DISPlay:COLor:PALette:POINt:INSert <PaletteName>, <PointIndex>

Inserts the entry at the specified index in the color table.

Setting parameters:
<PaletteName> color table

<PointIndex> row number in the color table

Usage: Setting only

DISPlay:COLor:PALette:POINt:REMove <PaletteName>, <PointIndex>

Removes the entry with the specified index from the color table.

Setting parameters:
<PaletteName> color table

<PointIndex> row number in the color table

Usage: Setting only

DISPlay:COLor:PALette:POINt[:VALue] <ColorTableName>, <Index>,<Position>, <Color>
DISPlay:COLor:PALette:POINt[:VALue]? <ColorTableName>, <Index>

Inserts a new entry or queries the specified entry in the specified color table.

Parameters:
<Position> Cumulative occurance value

Range: 0 to 100
Increment: 1
*RST: 50
Default unit: %

<Color> ARGB value of the color to be used for the table entry. ARGB=<Opacity(alpha) value><red value><green value><blue value>, in hexadecimal or decimal format.

Range: 0 to 4294967295
Increment: 1
*RST: 0

Parameters for setting and query:

<ColorTableName> Color table to be edited

<Index> Index (row number) of the new entry in the color table

Usage: Asynchronous command

DISPlay:COLor:PALette:POINt:COUNt? <PaletteName>

Queries the number of entries in the color table.

Query parameters:

<PaletteName> color table

Usage: Query only

24.7.2.3 Grid appearance

DISPlay:DIAGram:GRID 1300
DISPlay:DIAGram:LABels 1300
DISPlay:DIAGram:FINegrid 1301
DISPlay:DIAGram:CROSshair 1301
DISPlay:DIAGram:YFIXed 1301
DISPlay:DIAGram:XFIXed 1301

DISPlay:DIAGram:GRID <Show>

If enabled, a grid is displayed in the diagram area.

Parameters:

<Show> ON | OFF

Usage: Asynchronous command

DISPlay:DIAGram:LABels <ShowLabels>

If enabled, labels mark values on the x- and y-axes in specified intervals in the diagram.

Parameters:

<ShowLabels> ON | OFF

Usage: Asynchronous command

DISPlay:DIAGram:FINegrid <ShowFineScale>

If ON, the crosshair is displayed as a ruler with scale markers. If OFF, the crosshair is shown as dashed lines.

Parameters:
<ShowFineScale> ON | OFF

Usage: Asynchronous command

DISPlay:DIAGram:CROSshair <Crosshair>

If selected, a crosshair is displayed in the diagram area. A crosshair allows you to select a specific data point by its co-ordinates.

Parameters:
<Crosshair> ON | OFF

Usage: Asynchronous command

DISPlay:DIAGram:YFIXed <YGridFixed>

If enabled, the horizontal grid lines remain in their position when the position of the curve is changed. Only the values at the grid lines are adapted. This reflects the behavior of traditional oscilloscopes.

Parameters:
<YGridFixed> ON | OFF

Usage: Asynchronous command

DISPlay:DIAGram:XFIXed <XGridFixed>

If enabled, the vertical grid lines remain in their position when the horizontal position is changed. Only the values at the grid lines are adapted.

Parameters:
<XGridFixed> ON | OFF
*RST: OFF

Usage: Asynchronous command

24.7.2.4 Smartgrid

The following `LAYout` commands configure the SmartGrid: add and remove diagrams, and assign signals to the diagrams. In manual operation, you configure the SmartGrid by drag and drop.

LAYout:ADD 1302
LAYout:REMove 1304
LAYout:SHOW 1304
LAYout:SIGNal:ASSign 1304
LAYout:SIGNal:UNASsign 1306

LAYout:ADD <NodeName>, <ParentType>, <InsertBefore>, <FirstSource>, <DiagramName>

Adds a new diagram with a waveform on the screen, in relation to an existing diagram.

Setting parameters:

<NodeName> String with the name of the existing diagram

<ParentType> HORizontal | VERTical | TAB

Position of the new diagram in relation to the existing one.

HORizontal
Besides the existing diagram

VERTical
Above or below the existing diagram

TAB
In a new tab in the existing diagram

<InsertBefore> ON | OFF

If on, the new diagram is inserted to the left (for HORizontal), above (for VERTical) or in a tab in front the existing diagram.
HOR, ON = left to the existing diagram, defined in <NodeName>
HOR, OFF = right to the existing diagram
VERT, ON = above the existing diagram

VERT, OFF = below the existing diagram

<FirstSource> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | XY1 | XY2 | XY3 | XY4 | MRESult1 | MRESult2 | MRESult3 | MRESult4 | MRESult5 | MRESult6 | MRESult7 | MRESult8 | SBUS1 | SBUS2 | SBUS3 | SBUS4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | MSOB1 | MSOB2 | MSOB3 | MSOB4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2 | BATHtub1 | BMEas1 | SRESponse1 | SRHist1 | SDHist1 | TJHist1 | DJHist1 | DDJHist1 | PJHist1 | RJOHist1 | SRTRack1 | SDTRack1 | TJTRack1 | DJTRack1 | DDJTrack1 | PJTRack1 | RJOTrack1 | TJSPectrum1 | DDJSpectrum1 | PJSPectrum1 | RJOSpectrum1 | SEYE1 | BATHtub2 | BMEas2 | SRESponse2 | SRHist2 | SDHist2 | TJHist2 | DJHist2 | DDJHist2 | PJHist2 | RJOHist2 | SRTRack2 | SDTRack2 | TJTRack2 | DJTRack2 | DDJTrack2 | PJTRack2 | RJOTrack2 | TJSPectrum2 | DDJSpectrum2 | PJSPectrum2 | RJOSpectrum2 | SEYE2 | NBAThtub1 | NBMeas1 | TNHist1 | DNHist1 | DDNHist1 | PNHist1 | RNOHist1 | TNTRack1 | DNTRack1 | DDNTrack1 | PNTRack1 | RNOTrack1 | TNSPectrum1 | DDNSpectrum1 | PNSPectrum1 | RNOSpectrum1 | NBAThtub2 | NBMeas2 | TNHist2 | DNHist2 | DDNHist2 | PNHist2 | RNOHist2 | TNTRack2 | DNTRack2 | DDNTrack2 | PNTRack2 | RNOTrack2 | TNSPectrum2 | DDNSpectrum2 | PNSPectrum2 | RNOSpectrum2 | PJDHist1 | BUJHist1 | RJOHist1 | PJDTrack1 | PJDSpectrum1 | PJDHist2 | BUJHist2 | RJOHist2 | PJDTrack2 | PJDSpectrum2 | PNDHist1 | BUNHist1 | RNOHist1 | PNDTrack1 | PNDSpectrum1 | PNDHist2 | BUNHist2 | RNOHist2 | PNDTrack2 | PNDSpectrum2

Waveform to be displayed in the new diagram, see Chapter 24.4.2, "Waveform parameter", on page 1284.
Spectrum analysis, option R&S RTO6-K37: Spectrograms and timeline spectrums are automatically displayed in their own diagrams.
Eye diagram analysis with option R&S RTO6-K91: Eye diagrams are automatically displayed in their own diagrams.
TDR/TDT analysis with option R&S RTO6-K130: Measurement waveforms are automatically displayed in their own diagrams.
Advanced jitter analysis with option R&S RTO6-K133: Jitter analysis waveforms are automatically displayed in their own diagrams.
Advanced noise analysis with option R&S RTO6-K134 : Noise analysis waveforms are automatically displayed in their own diagrams.

<DiagramName> String with the name of the new diagram.

Example: `LAYout:ADD 'Diagram2',TAB,ON,C4W1,'MyDiagram3'`
Creates a new diagram 'MyDiagram3' with waveform C4W1 in a new tab that in front of 'Diagram2'.

Example: See Chapter 24.3.1.1, "Creating diagrams (SmartGrid)", on page 1260.

Usage: Setting only
Asynchronous command

LAYout:REMove <DiagramName>

Closes the specified diagram. The waveforms are displayed as minimized waveforms in their signal icons on the signal bar.

Setting parameters:
<DiagramName> String with the name of the diagram

Usage: Setting only
Asynchronous command

LAYout:SHOW <DiagramName>

Selects the specified diagram.

Setting parameters:
<DiagramName> String with the name of the diagram

Example: See Chapter 24.3.1.1, "Creating diagrams (SmartGrid)", on page 1260.

Usage: Setting only
Asynchronous command

LAYout:SIGNal:ASSign <DiagramName>, <Source>

Shows the specified waveform in the selected diagram.

Setting parameters:

<DiagramName> String with the diagram name

<Source> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | XY1 | XY2 | XY3 | XY4 | MRESult1 | MRESult2 | MRESult3 | MRESult4 | MRESult5 | MRESult6 | MRESult7 | MRESult8 | SBUS1 | SBUS2 | SBUS3 | SBUS4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | MSOB1 | MSOB2 | MSOB3 | MSOB4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2 | BATHtub1 | BMEas1 | SRESponse1 | SRHist1 | SDHist1 | TJHist1 | DJHist1 | DDJHist1 | PJHist1 | RJOHist1 | SRTRack1 | SDTRack1 | TJTRack1 | DJTRack1 | DDJTrack1 | PJTRack1 | RJOTrack1 | TJSPectrum1 | DDJSpectrum1 | PJSPectrum1 | RJOSpectrum1 | BATHtub2 | BMEas2 | SRESponse2 | SRHist2 | SDHist2 | TJHist2 | DJHist2 | DDJHist2 | PJHist2 | RJOHist2 | SRTRack2 | SDTRack2 | TJTRack2 | DJTRack2 | DDJTrack2 | PJTRack2 | RJOTrack2 | TJSPectrum2 | DDJSpectrum2 | PJSPectrum2 | RJOSpectrum2 | NBAThtub1 | NBMeas1 | TNHist1 | DNHist1 | DDNHist1 | PNHist1 | RNOHist1 | TNTRack1 | DNTRack1 | DDNTrack1 | PNTRack1 | RNOTrack1 | TNSPectrum1 | DDNSpectrum1 | PNSPectrum1 | RNOSpectrum1 | NBAThtub2 | NBMeas2 | TNHist2 | DNHist2 | DDNHist2 | PNHist2 | RNOHist2 | TNTRack2 | DNTRack2 | DDNTrack2 | PNTRack2 | RNOTrack2 | TNSPectrum2 | DDNSpectrum2 | PNSPectrum2 | RNOSpectrum2 | PJDHist1 | BUJHist1 | RJHist1 | PJDTrack1 | PJDSpectrum1 | PJDHist2 | BUJHist2 | RJHist2 | PJDTrack2 | PJDSpectrum2 | PNDHist1 | BUNHist1 | RNHist1 | PNDTrack1 | PNDSpectrum1 | PNDHist2 | BUNHist2 | RNHist2 | PNDTrack2 | PNDSpectrum2

Waveform to be assigned, see Chapter 24.4.2, "Waveform parameter", on page 1284
Spectrum analysis, option R&S RTO6-K37: Spectrograms and timeline spectrums are automatically displayed in their own diagrams.
Eye diagram analysis with option R&S RTO6-K91: Eye diagrams are automatically displayed in their own diagrams.
TDR/TDT analysis with option R&S RTO6-K130: Measurement waveforms are automatically displayed in their own diagrams.
Advanced jitter and noise analysis with option R&S RTO6-K133/K134

Example: See Chapter 24.3.1.1, "Creating diagrams (SmartGrid)", on page 1260.

Usage: Setting only
Asynchronous command

LAYout:SIGNal:UNASsign <Source>

Minimizes the specified waveform, i.e. removes it from the diagram and shows it in the signal icon.

Setting parameters:

<Source> See `LAYout:SIGNal:ASSign` on page 1304 and Chapter 24.4.2, "Waveform parameter", on page 1284

Example: See Chapter 24.3.1.1, "Creating diagrams (SmartGrid)", on page 1260.

Usage: Setting only
Asynchronous command

24.7.2.5 Diagram appearance

DISPlay:DIAGram:TITLe....1306
DISPlay:SIGBar[:STATe]....1306
DISPlay:GATE:TRANsparency....1306
LAYout:ZOOM:ONEDiagram....1307

DISPlay:DIAGram:TITLe <DiagTitleSt>

If enabled, the tab titles of all diagrams are displayed: "Diagram1", "Diagram2" ...

If disabled, the tab titles are not shown except for those in a tabbed diagram. In tabbed diagrams, the tab titles are required to change the tabs.

Parameters:

<DiagTitleSt> ON | OFF

Usage: Asynchronous command

DISPlay:SIGBar[:STATe] <State>

If enabled, the signal bar is displayed.

Parameters:

<State> ON | OFF

Usage: Asynchronous command

DISPlay:GATE:TRANsparency <Transparency>

Sets the transparency of the area that is defined as measurement or search gate.

Parameters:

<Transparency>
Range: 0 to 100
Increment: 1
*RST: 43
Default unit: %

Usage: Asynchronous command

Firmware/software: FW 3.20

LAYout:ZOOM:ONEDiagram <ShowInOne>

Shows all zooms of a diagram in one zoom window. The zoomed areas are overlaid for better comparison of the zoomed waveforms.

The command takes effect on all zoom diagrams.

Parameters:
<ShowInOne> ON | OFF
*RST: OFF

Usage: Asynchronous command

24.7.2.6 Annotations

To create an annotation (or waveform label), use `DISPlay:SIGNal:LABel:ADD`. Using the other `DISP:SIGN:LAB:...` commands, you can query the text and position of a label, and modify the initial settings. The <LabelID> and <Source> parameters identify each label uniquely. You can query the <LabelID> with `DISPlay:SIGNal:LABel:IDSList`.

DISPlay:SIGNal:LABel:POSMode....1307
DISPlay:SIGNal:LABel:FONTsize....1308
DISPlay:SIGNal:LABel:ADD....1308
DISPlay:SIGNal:LABel:REMove....1309
DISPlay:SIGNal:LABel:IDSList....1310
DISPlay:SIGNal:LABel:SHOW....1310
DISPlay:SIGNal:LABel:STATe....1311
DISPlay:SIGNal:LABel:TEXT....1312
DISPlay:SIGNal:LABel:HORizontal:ABSolute:POSition....1312
DISPlay:SIGNal:LABel:VERTical:ABSolute:POSition....1312
DISPlay:SIGNal:LABel:HORizontal:RELative:POSition....1313
DISPlay:SIGNal:LABel:VERTical:RELative:POSition....1313

DISPlay:SIGNal:LABel:POSMode <Source>, <PositionMode>
DISPlay:SIGNal:LABel:POSMode? <Source>

Modifies or queries the position mode: either relative to the diagram or with absolute values according to the units of the waveform. The position mode applies to all labels of the selected source. For different sources, different position modes can be selected.

Parameters:
<PositionMode> ABS | REL

ABS
Position in time and voltage values, or in other units depending on the waveform character. Absolute positions move with the waveform display when the scales, the vertical position or offset, or the reference point are changed.
Use `DISPlay:SIGNal:LABel:HORizontal:ABSolute:POSition` and `DISPlay:SIGNal:LABel:HORizontal:RELative:POSition` to set the position.

REL
Fixed label position in percent of the screen counting from the upper left corner.
Use `DISPlay:SIGNal:LABel:HORizontal:RELative:POSition` and `DISPlay:SIGNal:LABel:VERTical:RELative:POSition` to set the position.

Parameters for setting and query:

<Source> All waveforms that can be displayed, see `DISPlay:SIGNal:LABel:ADD`

Usage: Asynchronous command

DISPlay:SIGNal:LABel:FONTsize <Fontsize>

Sets the font size of waveform labels.

Parameters:

<Fontsize>
Range: 7 to 30
Increment: 1
*RST: 15

Usage: Asynchronous command

DISPlay:SIGNal:LABel:ADD <LabelID>, <Source>, <LabelText>, <PositionMode>, <XPositon>, <YPositon>

Creates a new waveform label for the specified source waveform.

Setting parameters:

<LabelID> String with the label identifier. The <LabelID> and <Source> parameters identify each label uniquely, so the label ID must be unique for the given waveform. Note the <LabelID> because it is not possible to query it, or to read it on the user interface.

<Source> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | XY1 | XY2 | XY3 | XY4 | MRESult1 | MRESult2 | MRESult3 | MRESult4 | MRESult5 | MRESult6 | MRESult7 | MRESult8 | QUICk | QUICK | SBUS1 | SBUS2 | SBUS3 | SBUS4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | MSOB1 | MSOB2 | MSOB3 | MSOB4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | SG1 | SG2 | SG3 | SG4 | SG5 | SG6 | SG7 | SG8 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL1 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Waveform to that the label belongs, see Chapter 24.4.1, "Waveform suffix", on page 1279.

<LabelText> String with the label text that is shown on the display

<PositionMode> ABS | REL

ABS
Position in time and voltage values, or in other units depending on the waveform character. Absolute positions move with the waveform display when the scales, the vertical position or offset, or the reference point are changed.

REL
Fixed label position in percent of the screen counting from the upper left corner.

<XPositon> Horizontal position of the label text. Values, range and unit depend on the position mode, the waveform position and scaling. For relative position mode, the range is 0 to 100 %.

<YPositon> Vertical position of the label text. Values, range and unit depend on the position mode, the waveform position and scaling. For relative position mode, the range is 0 to 100 %.

Usage: Setting only
Asynchronous command

DISPlay:SIGNal:LABel:REMove <LabelID>, <Source>

Deletes the specifies waveform label.

Setting parameters:

<LabelID> String with the label identifier.

<Source> All waveforms that can be displayed, see `DISPlay:SIGNal:LABel:ADD`

Example: `DISPlay:SIGNal:LABel:REMove 'Label1', C1W1`

Usage: Setting only
Asynchronous command

DISPlay:SIGNal:LABel:IDSList <Source>

Returns a list of label IDs for the specified source. Labels that are defined in the user interface get an automatic label "DiagramText<No>".

Parameters:
<Source> See `DISPlay:SIGNal:LABel:ADD`.

Usage: Asynchronous command

DISPlay:SIGNal:LABel:SHOW <Source>, <State>
DISPlay:SIGNal:LABel:SHOW? <Source>

Selects the source for the label display and displays all labels for that source that have state = ON in `DISPlay:SIGNal:LABel:ADD`.

Parameters:
<State> OFF | ON

Parameters for setting and query:

<Source> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | XY1 | XY2 | XY3 | XY4 | MRESult1 | MRESult2 | MRESult3 | MRESult4 | MRESult5 | MRESult6 | MRESult7 | MRESult8 | SBUS1 | SBUS2 | SBUS3 | SBUS4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | MSOB1 | MSOB2 | MSOB3 | MSOB4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2 | BATHtub1 | BMEas1 | SRESponse1 | SRHist1 | SDHist1 | TJHist1 | DJHist1 | DDJHist1 | PJHist1 | RJOHist1 | SRTRack1 | SDTRack1 | TJTRack1 | DJTRack1 | DDJTrack1 | PJTRack1 | RJOTrack1 | TJSPectrum1 | DDJSpectrum1 | PJSPectrum1 | RJOSpectrum1 | SEYE1 | BATHtub2 | BMEas2 | SRESponse2 | SRHist2 | SDHist2 | TJHist2 | DJHist2 | DDJHist2 | PJHist2 | RJOHist2 | SRTRack2 | SDTRack2 | TJTRack2 | DJTRack2 | DDJTrack2 | PJTRack2 | RJOTrack2 | TJSPectrum2 | DDJSpectrum2 | PJSPectrum2 | RJOSpectrum2 | SEYE2 | NBAThtub1 | NBMeas1 | TNHist1 | DNHist1 | DDNHist1 | PNHist1 | RNOHist1 | TNTRack1 | DNTRack1 | DDNTrack1 | PNTRack1 | RNOTrack1 | TNSPectrum1 | DDNSpectrum1 | PNSPectrum1 | RNOSpectrum1 | NBAThtub2 | NBMeas2 | TNHist2 | DNHist2 | DDNHist2 | PNHist2 | RNOHist2 | TNTRack2 | DNTRack2 | DDNTrack2 | PNTRack2 | RNOTrack2 | TNSPectrum2 | DDNSpectrum2 | PNSPectrum2 | RNOSpectrum2 | PJDHist1 | BUJHist1 | RJOHist1 | PJDTrack1 | PJDSpectrum1 | PJDHist2 | BUJHist2 | RJOHist2 | PJDTrack2 | PJDSpectrum2 | PNDHist1 | BUNHist1 | RNOHist1 | PNDTrack1 | PNDSpectrum1 | PNDHist2 | BUNHist2 | RNOHist2 | PNDTrack2 | PNDSpectrum2

Usage: Asynchronous command

DISPlay:SIGNal:LABel:STATe <LabelID>, <Source>, <State>
DISPlay:SIGNal:LABel:STATe? <LabelID>, <Source>

Displays or hides a single label that is identified by the signal source and label ID.

Parameters:

<State> OFF | ON

ON: shows the label on the screen.
`DISPlay:SIGNal:LABel:SHOW` must be set to ON for the source.

Parameters for setting and query:

<LabelID> String with the label identifier.

<Source> See `DISPlay:SIGNal:LABel:ADD`.

Usage: Asynchronous command

DISPlay:SIGNal:LABel:TEXT <LabelID>, <Source>, <LabelText>
DISPlay:SIGNal:LABel:TEXT? <LabelID>, <Source>

Modifies or queries the text of the specified label.

Parameters:
<LabelText> String with the label text that is shown

Parameters for setting and query:
<LabelID> String with the label identifier.

<Source> All waveforms that can be displayed, see `DISPlay:SIGNal:LABel:ADD`

Usage: Asynchronous command

DISPlay:SIGNal:LABel:HORizontal:ABSolute:POSition <LabelID>, <Source>, <Position>
DISPlay:SIGNal:LABel:HORizontal:ABSolute:POSition? <LabelID>, <Source>
DISPlay:SIGNal:LABel:VERTical:ABSolute:POSition <LabelID>, <Source>, <Position>
DISPlay:SIGNal:LABel:VERTical:ABSolute:POSition? <LabelID>, <Source>

Modifies or queries the absolute horizontal and vertical positiions of the specified label if `DISPlay:SIGNal:LABel:POSMode` is set to `ABS`.

Parameters:
<Position> Range: Depends on waveform position and scaling
Default unit: s and V, or in other units depending on the waveform character

Parameters for setting and query:
<LabelID> String with the label identifier.

<Source> All waveforms that can be displayed, see `DISPlay:SIGNal:LABel:ADD`

Example:

```
:DISPlay:SIGNal:LABel:POSMode C2W1, ABS
DISPlay:SIGNal:LABel:HORizontal:ABSolute:POSition 'Label1', C2W1, 5e-09
DISPlay:SIGNal:LABel:VERTical:ABSolute:POSition 'Label1', C2W1, -0.1
```

Move the label to 5 ns and -0.1 V.

DISPlay:SIGNal:LABel:HORizontal:RELative:POSition <LabelID>, <Source>, <Position>
DISPlay:SIGNal:LABel:HORizontal:RELative:POSition? <LabelID>, <Source>
DISPlay:SIGNal:LABel:VERTical:RELative:POSition <LabelID>, <Source>, <Position>
DISPlay:SIGNal:LABel:VERTical:RELative:POSition? <LabelID>, <Source>

Modifies or queries the relative horizontal and vertical positiions of the specified label if `DISPlay:SIGNal:LABel:POSMode` is set to `REL`.

Parameters:

<Position> Position in percent of the screen counting from the upper left corner.

Range: 0 to 100
Default unit: %

Parameters for setting and query:

<LabelID> String with the label identifier.

<Source> All waveforms that can be displayed, see `DISPlay:SIGNal:LABel:ADD`

Example:

```
:DISPlay:SIGNal:LABel:POSMode C2W1, REL
DISPlay:SIGNal:LABel:HORizontal:RELative:POSition 'Label1', C2W1, 30
DISPlay:SIGNal:LABel:VERTical:RELative:POSition 'Label1', C2W1, 70
```

Move the label to new relative position: horizontal at 30 % and vertical at 70 % of the screen.

24.7.2.7 Dialog appearance

DISPlay:EXTended:PORDialogs <PopOutResDialogs>

If an external monitor is connected to the instrument, enables the display of result boxes on the external monitor.

Parameters:

<PopOutResDialogs> ON | OFF

*RST: OFF

DISPlay:EXTended:POSDialogs <PopOutSettDialogs>

If an external monitor is connected to the instrument, enables the display of dialog boxes on the external monitor.

Parameters:

<PopOutSettDialogs> ON | OFF

*RST: OFF

24.7.2.8 Cursor appearance

CURSor<m>:STYLe <Style>

Defines how the cursor is displayed in the diagram.

Suffix:

<m> 1..4
Selects the cursor set.

Parameters:

<Style> LINes | LRHombus | VLRHombus | RHOMbus

LINes
The cursors are displayed as lines.

LRHombus
The cursors are displayed as lines. The intersections of the cursors with the waveforms are displayed by rhombus-shaped points.

VLRHombus
The cursors are displayed only as vertical lines. The intersections of the cursors with the waveforms are displayed by rhombus-shaped points.

RHOMbus
The intersections of the cursors with the waveforms are displayed by rhombus-shaped points.

*RST: LINes

Usage: Asynchronous command

DISPlay:RESultboxes:CUPosition <Position>

Defines the position of the cursor measurement results on the screen.

Parameters:

<Position> PREV | FLOA | DOCK

PREV
Preview: result icon on the signal bar.

FLOA
Floating result box in front of the diagrams.

DOCK
Docked: fixed tab below the diagrams.

*RST: DOCK

Usage: Asynchronous command

24.7.2.9 Measurement appearance

MEASurement<m>:DISPlay:GROuping <GroupResDialogs>

If ON, all measurement results are shown in one result box. If disabled, the results of each measurement group are shown in a separate result box. The default position is ignored.

Suffix:
<m> 1..10
The suffix is irrelevant. The setting affects all measurements except for the peak list.

Parameters:
<GroupResDialogs> ON | OFF
*RST: ON

Usage: Asynchronous command

DISPlay:RESultboxes:MEPosition <Position>

Defines the position of the measurement results on the display.

Parameters:
<Position> PREV | FLOA | DOCK

PREV
Preview: result icon on the signal bar

FLOA
Floating: floating result box in front of the diagrams

DOCK
Docked: fixed tab below the diagrams

*RST: DOCK

Usage: Asynchronous command

24.7.2.10 Peak list appearance

MEASurement<m>:RESult:LABorder <MeasType>, <FrameType>
MEASurement<m>:RESult:LABorder? <MeasType>

Defines the layout of the labels.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<FrameType> NOBorder | UNDERLINE | FULL

NOBorder
Label without border or lines.

UNDERLINE
The label is underlined.

FULL
The label is surrounded by a frame.

Parameters for setting and query:

<MeasType> PLISt

Only for peaklist measurements. For other measurements, this command returns an error.

MEASurement<m>:RESult:INVerse <MeasType>, <State>
MEASurement<m>:RESult:INVerse? <MeasType>

Displays labels with black font on white background using the "Full frame" label type (if `MEASurement<m>:RESult:LABorder=ON`).).

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<State> ON | OFF

Parameters for setting and query:

<MeasType> PLISt

Only for peaklist measurements. For other measurement, this command returns an error.

Example: Display labels.

```
MEAS:RES:SHL PLIS,ON
```

Select inverted labels.

```
MEAS:RES:INV PLIS,ON
```

Query the type of labels for peak lists.

```
MEAS:RES:INV? PLIS
//Result: ON
```

24.7.2.11 Mask appearance

MTESt:LABel....1317
MTESt:RERename....1317
MTESt:HIGHlight:INFinite....1317
MTESt:HIGHlight:STATe....1317
MTESt:HIGHlight:TIME....1317
MTESt:COLor:UNMatch....1318
MTESt:COLor:MATCh....1318
MTESt:COLor:BORDer....1318
MTESt:COLor:INTerior....1319

MTESt:LABel <LabelState>

Switches the display of the mask test name on or off.

To change the name of the mask test, use `MTESt:REName` on page 1317.

Parameters:
<LabelState> ON | OFF
*RST: ON

Usage: Asynchronous command

MTESt:REName <MaskTestName>, <NewName>
MTESt:REName? <MaskTestName>

Changes the name of the mask test.

Parameters:
<NewName> String with the new mask test name.

Parameters for setting and query:
<MaskTestName> String with the existing mask test name.

MTESt:HIGHlight:INFinite <State>

If ON, the mask hits are highlighted for an unlimited period of time.

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

MTESt:HIGHlight:STATe <HighlightHits>

If ON, the mask hits are highlighted on the screen. You can define the color and the time of the hit display.

Parameters:
<HighlightHits> ON | OFF
*RST: ON

Usage: Asynchronous command

MTESt:HIGHlight:TIME <HighlightTime>

Sets the time how long the mask hits are highlighted.

Parameters:

<HighlightTime>
Range: 0.05 to 50
Increment: 0.1
*RST: 1
Default unit: s

Usage: Asynchronous command

MTESt:COLor:UNMatch <ClrUnmatchedMsk>

Sets the color of masks segments that were not hit.

Parameters:

<ClrUnmatchedMsk>
Decimal value corresponding to the ARGB hex value that is shown in the "Color" dialog box. ARGB is a 32-bit unsigned integer with alpha, red, green, blue word order. Alpha defines the transparency.
To get the value, covert the hex value from the dialog to decimal value.

Range: 0 (black) to 4294967295 (white)
Increment: 1
*RST: 2578098858

Usage: Asynchronous command

MTESt:COLor:MATCh <ClrMatchedPixel>

Sets the color of samples that violated the mask.

Parameters:

<ClrMatchedPixel>
Decimal value corresponding to the ARGB hex value that is shown in the "Color" dialog box. ARGB is a 32-bit unsigned integer with alpha, red, green, blue word order. Alpha defines the transparency.
To get the value, covert the hex value from the dialog to decimal value.

Range: 0 (black) to 4294967295 (white)
Increment: 1
*RST: 4294901760

Usage: Asynchronous command

MTESt:COLor:BORDer <ClrMatchedMskBorder>

Sets the color of masks segments that were touched at the border. In this case, the resolution is not sufficient to detect if the mask was really hit or not. Zoom into the concerned area to see the actual result.

Parameters:

<ClrMatchedMskBorder> Decimal value corresponding to the ARGB hex value that is shown in the "Color" dialog box. ARGB is a 32-bit unsigned integer with alpha, red, green, blue word order. Alpha defines the transparency.
To get the value, covert the hex value from the dialog to decimal value.

Range: 0 (black) to 4294967295 (white)
Increment: 1
*RST: 2580309606

Usage: Asynchronous command

MTESt:COLor:INTerior <ClrMatchedMskInterior>

Sets the color of mask segments the signal has entered into.

Parameters:

<ClrMatchedMskInterior> Decimal value corresponding to the ARGB hex value that is shown in the "Color" dialog box. ARGB is a 32-bit unsigned integer with alpha, red, green, blue word order. Alpha defines the transparency.
To get the value, covert the hex value from the dialog to decimal value.

Range: 0 (black) to 4294967295 (white)
Increment: 1
*RST: 2578055168

Usage: Asynchronous command

24.7.3 Display

24.7.3.1 Persistence

DISPlay:PERSistence[:STATe]....1319
DISPlay:PERSistence:INFinite....1320
DISPlay:PERSistence:TIME....1320
DISPlay:PERSistence:RESet....1320

DISPlay:PERSistence[:STATe] <State>

If enabled, each new data point in the diagram area remains on the screen for the duration defined using `DISPlay:PERSistence:TIME`, or as long as `DISPlay:PERSistence:INFinite` is enabled.

If disabled, the signal value is only displayed as long as it actually occurs.

Parameters:
<State> ON | OFF
*RST: ON

Usage: Asynchronous command

DISPlay:PERSistence:INFinite <State>

If persistence is enabled (`DISPlay:PERSistence[:STATe]`), each new data point in the diagram area remains on the screen infinitely until this command is set to "OFF".

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

DISPlay:PERSistence:TIME <Time>

If persistence is enabled (`DISPlay:PERSistence[:STATe]`), each new data point in the diagram area remains on the screen for the duration defined here.

Parameters:
<Time> Range: 0.05 to 50
Increment: 0.05
*RST: 0.05
Default unit: s

Usage: Asynchronous command

DISPlay:PERSistence:RESet

Resets the display, removing persistent values.

Usage: Event
Asynchronous command

24.7.3.2 Signal

DISPlay:INTensity <Intensity>

This value determines the strength of the waveform line in the diagram. Enter a percentage between 0 (not visible) and 100% (very strong).

The exact mapping of the cumulative value occurences according to the assigned color table is guaranteed only if the intensity is set to 50% (default). All other intensity values falsify the mapping but may improve the visibility of the signal.

Parameters:

<Intensity> Range: 0 to 100
Increment: 1
*RST: 50
Default unit: %

Usage: Asynchronous command

DISPlay:DIAGram:STYLe <Style>

Select the style in which the waveform is displayed.

Parameters:

<Style> VECTors | DOTS

VECTors
The individual data points are connected by a line.

DOTS
Only the individual data points are displayed.

*RST: VECTors

Usage: Asynchronous command

24.7.3.3 Clear results

DISPlay:CLR

Deletes all measurement results including long term measurement and statistics, all waveforms, and the history.

Usage: Event
Asynchronous command

24.7.4 User-defined preset

USRDefined:RST:NAME <Name>

Sets the path and file name of a user-defined preset file.

Parameters:

<Name> String parameter

USRDefined:RST:OPEN

opens and loads a user-defined preset file. Name and path of the preset file are defined with `USRDefined:RST:NAME`.

Usage: Event

USRDefined:RST:SAVE

Saves the current instrument settings to a user-defined preset file. Name and path of the preset file are defined with `USRDefined:RST:NAME`.

Usage: Event

USRDefined:RST[:ENABle] <Name>

If ON, the settings from a previously defined preset file are restored when the [Preset] key is pressed. The preset file is defined with `USRDefined:RST:NAME`.

If OFF, [Preset] sets the instrument to the factory defaults.

Parameters:
<Name> ON | OFF
*RST: OFF

24.7.5 Save / recall

See the following chapters:

- Chapter 24.16.10, "Autonaming", on page 1666
- Chapter 24.16.11, "CSV export", on page 1668
- Chapter 24.16.12, "External application", on page 1668

24.7.6 Maintenance

24.7.6.1 Self-alignment

CALibration:DATE?.....1322
CALibration:TIME?.....1322
CALibration:RESult?.....1323

CALibration:DATE?

Returns the date of the last selfalignment.

Return values:
<Date>

Usage: Query only
Asynchronous command

CALibration:TIME?

Returns the time of the last selfalignment.

Return values:
<Time>

Usage: Query only
Asynchronous command

CALibration:RESult?

Returns the result of the last selfalignment and the current alignment status. In remote mode, `*CAL?` provides more detailed information.

Return values:
<ResultState> PASSed | FAILed | NOALigndata
*RST: FAILed

Usage: Query only
Asynchronous command

24.7.6.2 Self-test

DIAGnostic:SERVice:STST:STATe?

Returns the summary result of the selftest.

Return values:
<State> PSSD | FAILed | UNDefined
*RST: UNDefined

Usage: Query only
Asynchronous command

24.7.6.3 Hardware

SYSTem:DFPRint [<Path>]

The device footprint contains the configuration of the instrument, installed modules, installed software and software licenses. This information is written in the device footprint xml file might be useful in case of maintenance or support request.

The query returns the information as block data. The setting command saves the device footprint xml file in the specified path.

It is also possible to access the device footprint xml file via the instrument's web browser. Therefore, the directory containing the xml file must be enabled for sharing.

Setting parameters:
<Path> String parameter, specifying the target path of the footprint file.

Return values:
<DeviceFootprint> Content of the device footprint xml file as block data

DIAGnostic:SERVice:FWVersion?

Returns the firmware version that is currently installed on the instrument.

Return values:
<FirmwareVersion> Version string

Usage: Query only
Asynchronous command

DIAGnostic:SERVice:PARTnumber <Number>

Returns the part number (order number) of your instrument. This number is required to order a new option, and in case of service.

Parameters:
<Number> Number string

Usage: Asynchronous command

DIAGnostic:SERVice:SERialnumber?

Returns the serial number of your instrument. This number is required to order a new option, and in case of service.

Return values:
<SerialNumber> Number string

Usage: Query only
Asynchronous command

DIAGnostic:SERVice:CHANnelcount?

Queries the number of available channels.

Return values:
<ChannelCount> Range: 0 to 4
Increment: 1
*RST: 0

Usage: Query only
Asynchronous command

Firmware/software: V 2.00

24.8 Acquisition and setup

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

- Starting and stopping acquisition 1325
- Time base 1326
- Acquisition 1332
- Fast segmentation 1336
- Vertical 1337
- Waveform data 1342
- Probes 1343
- R&S RT-ZVC probe 1366
- Differential signals 1379
- Digital filter 1382
- Skew 1384
- AUX OUT 1385
- High definition mode 1386
- Reference clock 1388

24.8.1 Starting and stopping acquisition

RUNContinous 1325
RUN 1325
RUNSingle 1325
SINGle 1325
STOP 1326

RUNContinous
RUN

Starts the continuous acquisition.

Usage: Event
Asynchronous command

RUNSingle
SINGle

Starts a defined number of acquisition cycles. The number of cycles is set with `ACQuire:COUNt`.

Usage: Event
Asynchronous command

STOP

Stops the running acquistion.

Usage: Event
Asynchronous command

24.8.2 Time base

TIMebase:SCALe....1326
TIMebase:RANGe....1327
TIMebase:DIVisions?....1327
TIMebase:HORizontal:POSition....1327
TIMebase:REFerence....1327
TRIGger<m>:OFFSet:LIMited....1328
AUToscale....1328
ACQuire:POINts:AUTO....1328
ACQuire:POINts:AADJust....1329
ACQuire:POINts:MAXimum....1329
ACQuire:POINts:ARATe?....1329
ACQuire:SRATe....1329
ACQuire:SRReal....1330
ACQuire:RESolution....1330
ACQuire:POINts[:VALue]....1330
TIMebase:ROLL:ENABle....1331
TIMebase:ROLL:STATe?....1331
TIMebase:ROLL:MTIMe....1331

TIMebase:SCALe <TimeScale>

Sets the horizontal scale - the time per division on the x-axis - for all channel and math waveforms.

The setting accuracy depends on the current resolution (sample rate).

- No interpolation:
  The resolution is an integer multiple of the ADC sample rate.
- With interpolation:
  Any value for the horizontal scale can be set.

Parameters:

<TimeScale>
Range: 25E-12 to 10000 (RTO, RTP) | 5000 (RTE)
Increment: 1E-12
*RST: 10E-9
Default unit: s/div

Usage: Asynchronous command

TIMebase:RANGe <AcquisitionTime>

Defines the time of one acquisition, that is the time across the 10 divisions of the diagram: *TimeScale*10*.

Parameters:

<AcquisitionTime>
Range: 250E-12 to 100E+3 (RTO, RTP) | 50E+3 (RTE)
Increment: 1E-12
*RST: 0.5
Default unit: s

Usage: Asynchronous command

TIMebase:DIVisions?

Queries the number of horizontal divisions on the screen. The number cannot be changed.

Return values:

<HorizDivCnt>
Range: 4 to 20
Increment: 2
*RST: 10

Usage: Query only
Asynchronous command

TIMebase:HORizontal:POSition <RescalCenterTime>

Defines the time distance between the reference point and the trigger point (the zero point of the diagram). The reference point marks the rescaling center of the time scale.

Parameters:

<RescalCenterTime>
Range: -100E+24 to 100E+24
Increment: 1E-12
*RST: 0
Default unit: s

Usage: Asynchronous command

TIMebase:REFerence <RescaleCtrPos>

Sets the position of the reference point in % of the screen. The reference point marks the rescaling center of the time scale. If you modify the time scale, the reference point remains fixed on the screen, and the scale is stretched or compresses to both sides of the reference point.

Parameters:

<RescaleCtrPos>
Range: 0 to 100
Increment: 1
*RST: 50
Default unit: %

Usage: Asynchronous command

TRIGger<m>:OFFSet:LIMited <State>

If ON, the horizontal position cannot be set outside the visible waveform diagram.

See also: `TIMebase:HORizontal:POSition` on page 1327

Suffix:

<m> 1..3
The numeric suffix is irrelevant.

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

AUToscale

Performs an autoset process: analyzes the enabled channel signals, and obtains appropriate horizontal, vertical, and trigger settings to display stable waveforms.

Rohde & Schwarz does not recommend using the autoset in remote control. To adjust the oscilloscope remotely, especially for automated testing applications, use the remote commands that adjust the horizontal, vertical and trigger settings."

Usage: Event
Asynchronous command

ACQuire:POINts:AUTO <ReclgthManual>

Selection to keep constant either the resolution or the record length when you adjust the time scale (`TIMebase:SCALe`) or acquisition time (`TIMebase:RANGe`).

Parameters:

<ReclgthManual> RESolution | RECLength

RESolution
Resolution is kept constant. Set the required resolution value with `ACQuire:RESolution`.

RECLength
The record length is kept constant. Set the required record length value with `ACQuire:POINts[:VALue]`.

*RST: RESolution

Usage: Asynchronous command

ACQuire:POINts:AADJust <AutoAdjust>

Prevents undersampling and ensures a sufficient resolution to acquire the correct waveform if the time scale is changed. The setting takes effect if the changed parameter - resolution or record length - reaches a limit. The instrument automatically keeps this parameter constant at its limit, and changes the other parameter regardless of the `ACQuire:POINts:AUTO` setting.

Parameters:

<AutoAdjust> ON | OFF
*RST: ON

Usage: Asynchronous command

ACQuire:POINts:MAXimum <ReclgthLim>

Sets a limit for the record length to prevent very large records. This value only takes effect if a constant resolution is selected with `ACQuire:POINts:AUTO`. If you increase the time scale, the resolution remains constant and the record length increases until the limit is reached. Further increase of the time scale changes the resolution and keeps the record length limit.

Parameters:

<ReclgthLim> Range: 1000 to 800 MSa. The actual maximum can be lower depending on the installed options, number of active channels, measurements and math waveforms.
Increment: 2
*RST: 10E+6
Default unit: pts

Usage: Asynchronous command

ACQuire:POINts:ARATe?

Retrieves the sample rate of the ADC, that is the number of points that are sampled by the ADC in one second.

Return values:

<ADCSampleRate> Range: 10 Gsample/s
Default unit: Sa/s

Usage: Query only
Asynchronous command

ACQuire:SRATe <SampleRate>

Defines the sample rate, that is the number of recorded waveform samples per second.

Parameters:

<SampleRate>
Range: 10 Gsample/s
Increment: 1
*RST: 10 Gsample/s
Default unit: Sa/s

Usage: Asynchronous command

ACQuire:SRReal <RealSampleRate>

Sets the number of captured waveform points per second. It considers the samples of the ADC, and the reduction of waveform points by decimation.

If interpolation is not active, the sample rate is the reciprocal value of the resolution and thus also depends on the acquisition time and the record length.

If interpolation is active, the sample rate is limited to the ADC sample rate.

Parameters:

<RealSampleRate>
Range: 10 Gsample/s
Increment: 1
*RST: 10 Gsample/s
Default unit: Sa/s

Usage: Asynchronous command

Firmware/software: FW 3.40

ACQuire:RESolution <Resolution>

Indicates the time between two waveform points in the record.

Parameters:

<Resolution>
A fine resolution with low values produces a more precise waveform record.
Range: 1E-15 to 0.5
Increment: 10E-12
*RST: 100E-12
Default unit: s

Usage: Asynchronous command

ACQuire:POINts[:VALue] <RecordLength>

Indicates the record length, the number of recorded waveform points that build the waveform across the acquisition time. `[:VALue]` can be omitted.

Parameters:

<RecordLength> Number of recorded waveform points.

Range: 1000 to 1000000000
Increment: 2
*RST: 1000
Default unit: pts

Usage: Asynchronous command

TIMebase:ROLL:ENABle <Mode>

Activates the automatic roll mode.

Parameters:

<Mode> AUTO | OFF

AUTO: the instrument activates the roll mode under specific conditions.
See also "Roll mode" on page 121.

*RST: AUTO

Usage: Asynchronous command

TIMebase:ROLL:STATe?

Returns the status of the roll mode.

Return values:

<State> ON | OFF

*RST: OFF

Usage: Query only
Asynchronous command

TIMebase:ROLL:MTIMe <MinHorizGain>

The roll mode is enabled automatically if the acquisition time exceeds the given value, and if `TIMebase:ROLL:ENABle` is set to AUTO.

Parameters:

<MinHorizGain> Treshold value for roll mode enabling.

Range: 1 to 600
Increment: 1
*RST: 10
Default unit: s

Usage: Asynchronous command

24.8.3 Acquisition

ACQuire:INTerpolate 1332
ACQuire:CDTA 1332
ACQuire:MUWaveform 1333
CHANnel<m>[:WAVeform<n>][:STATe] 1333
CHANnel<m>[:WAVeform<n>]:TYPE 1333
CHANnel<m>[:WAVeform<n>]:ARIThmetics 1334
ACQuire:COUNt 1334
ACQuire:ARESet:IMMediate 1335
ACQuire:ARESet:MODE 1335
ACQuire:ARESet:TIME 1335
ACQuire:ARESet:COUNt 1335

ACQuire:INTerpolate <IntpolMd>

Selects the interpolation method.

See also: "Interpolation" on page 128.

Parameters:

<IntpolMd> LINear | SINX | SMHD

LINear
Linear interpolation between two adjacent sample points

SINX
Interpolation by means of a sin(x)/x curve.

SMHD
Sample/Hold causes a histogram-like interpolation.

*RST: SINX

Usage: Asynchronous command

ACQuire:CDTA <CoupleAcquSet>

Sets the acquisition mode and the waveform arithmetic of all channels to the last set value.

If the acquisition settings are coupled, `ACQuire:MUWaveform` is not relevant, only one waveform per channel can be used.

Parameters:

<CoupleAcquSet> ON | OFF

*RST: ON

Usage: Asynchronous command

Firmware/software: FW 3.30
Substitutes `ACQuire:CMODode`

ACQuire:MUWaveform <MultiWaveform>

For each channel, up to three waveforms can be shown and analyzed. The decimation mode and the waveform arithmetic are specific for each waveform. So you can analyze several aspects of the signal: For example, waveform1 shows the peaks, and waveform2 shows the average of the signal.

Parameters:

<MultiWaveform> ON | OFF

*RST: OFF

Usage: Asynchronous command

Firmware/software: FW 3.20

CHANnel<m>[:WAVeform<n>][:STATe] <State>

Activates or deactivates a waveform. `[:STATe]` can be omitted.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

CHANnel<m>[:WAVeform<n>]:TYPE <DecimationMode>

Selects the method to reduce the data stream of the ADC to a stream of waveform points with lower sample rate.

See also:"Decimation" on page 129.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Parameters:

<DecimationMode> SAMPle | PDETect | HRESolution | RMS

SAMPle
One of n samples in a sample interval of the ADC is recorded as waveform point.

PDETect
Peak Detect: the minimum and the maximum of n samples in a sample interval are recorded as waveform points.

HRESolution
High resolution: The average of n sample points is recorded as waveform point.

RMS
The waveform point is the root mean square of n sample values.

*RST: SAMPle

Usage: Asynchronous command

CHANnel<m>[:WAVeform<n>]:ARIThmetics <Arithmetics>

Selects the method to build the resulting waveform from several consecutive acquisitions of the signal. To define the number of acquisitions, use `ACQuire:COUNt`.

See also: "Arithmetic" on page 130.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Parameters:

<Arithmetics> OFF | ENVelope | AVERage

OFF
The data of the current acquisition is recorded according to the decimation settings.

ENVelope
Detects the minimum and maximum values in an sample interval over a number of acquisitions. To define the reset method, use ...

AVERage
Calculates the average from the data of the current acquisition and a number of acquisitions before.

*RST: OFF

Usage: Asynchronous command

ACQuire:COUNt <MaxAcqCnt>

The acquisition and average count has a double effect:

- It sets the number of waveforms acquired with `RUNSingle`.
- It defines the number of waveforms used to calculate the average waveform.

Parameters:

<MaxAcqCnt> Range: 1 to 16777215
Increment: 10
*RST: 1

Usage: Asynchronous command

ACQuire:ARESet:IMMediate

Forces the immediate restart of the envelope and average calculation for all waveforms.

Usage: Event
Asynchronous command

ACQuire:ARESet:MODE <ArtmRst>

Defines when the envelope and average evaluation restarts.

Parameters:

<ArtmRst> NONE | TIME | WFMS

TIME
Restarts the envelope and average calculation after the time defined with `ACQuire:ARESet:TIME`.

WFMS
Restarts the envelope and average calculation after a number of acquired waveforms defined with `ACQuire:ARESet:COUNt` on page 1335.

*RST: NONE

Usage: Asynchronous command

ACQuire:ARESet:TIME <EnvelopeTimeout>

Defines the time after which the envelope and average evaluation restarts.

The setting is relevant if `ACQuire:ARESet:MODE` is set to `TIME`.

Parameters:

<EnvelopeTimeout> Range: 0.1 to 10000
Increment: 0.01
*RST: 0.1
Default unit: s

Usage: Asynchronous command

ACQuire:ARESet:COUNt <NofWaveforms>

Defines the number of acquired waveforms after which the envelope and average evaluation restarts.

The setting is relevant if `ACQuire:ARESet:MODE` is set to `WFMS`.

Parameters:

<NofWaveforms> Range: 2 to 16777215
Increment: 10
*RST: 1000

Usage: Asynchronous command

24.8.4 Fast segmentation

ACQuire:SEGMented:STATe 1336
ACQuire:SEGMented:MAX 1336
ACQuire:SEGMented:AUToreplay 1336

ACQuire:SEGMented:STATe <State>

Switches the fast segmentation mode on and off.

Parameters:

<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

ACQuire:SEGMented:MAX <MaxAcquisitions>

The number of acquisitions in a fast segmentation acquisition series depends on the record length.

Parameters:

<MaxAcquisitions> ON | OFF

ON
Acquires the maximum possible number of acquisitions in a series.

OFF
Acquires the number of acquisitions defined using `ACQuire:COUNt`.

*RST: OFF

Usage: Asynchronous command

ACQuire:SEGMented:AUToreplay <ReplayAfterAcq>

If enabled, the instrument starts processing and displaying the data as soon as the acquisition series is captured completely. Depending on the number of acquisitions, it may take some time until the acquisition series is displayed. If the setting is disabled, the instrument only captures the data and stores it in the sample memory.

Parameters:
<ReplayAfterAcq> ON | OFF
*RST: ON

Usage: Asynchronous command

24.8.5 Vertical

CHANnel<m>:STATe 1337
CHANnel<m>:COUPling 1337
CHANnel<m>:GND 1338
CHANnel<m>:SCALe 1338
CHANnel<m>:RANGe 1338
CHANnel<m>:POSition 1339
CHANnel<m>:OFFSet 1339
CHANnel<m>:INVert 1339
CHANnel<m>:BANDwidth 1340
CHANnel<m>:CPLing 1340
CHANnel<m>:IMPedance 1340
CHANnel<m>:OVERload 1341
CHANnel<m>:FILDesign 1341

CHANnel<m>:STATe <State>

Switches the channel signal on or off.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

CHANnel<m>:COUPling <Value>

Selects the connection of the indicated channel signal.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<Value> DC | DCLimit | AC

DC
Direct connection with 50 Ω termination, passes both DC and AC components.

DCLimit
Direct connection with 1 MΩ termination, passes both DC and AC components.

AC
Connection with 1 MΩ termination, removes DC and very low-frequency components.

CHANnel<m>:GND <State>

Connects the signal to the ground.

Suffix:
<m> 1..4

Parameters:
<State> ON | OFF
*RST: OFF

CHANnel<m>:SCALe <Scale>

Sets the vertical scale for the indicated channel.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<Scale> Scale value, given in Volts per division.
Range: Depends on attenuation factors, coupling, and instrument model, see below.
Increment: Depends on vertical and probe settings
*RST: 0.05
Default unit: V/div

Usage: Asynchronous command

CHANnel<m>:RANGe <Range>

Sets the voltage range across the 10 vertical divisions of the diagram. Use the command alternativly instead of `CHANnel<m>:SCALe`.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<Range> Voltage range value
Range: Dependencies see below.
Increment: Depends on vertical and probe settings
*RST: 0.5
Default unit: V/div

Usage: Asynchronous command

CHANnel<m>:POSition <Position>

Sets the vertical position of the indicated channel as a graphical value.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<Position> Positive values move the waveform up, negative values move it down.

Range: -5 to 5
Increment: 0.01
*RST: 0
Default unit: div

Usage: Asynchronous command

CHANnel<m>:OFFSet <Offset>

The offset voltage is subtracted to correct an offset-affected signal. The offset of a signal is determined and set by the autoset procedure.

See also: "Offset" on page 139.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<Offset> Negative values move the waveform up, positive values move it down.

Range: Depends on attenuation factors, input coupling, and the offset compensation range of active probes. The nominal offset range for 1:1 attenuation and probe offset compensation = 0 is specified in the data sheet.
Increment: Depends on vertical and probe settings
*RST: 0
Default unit: V

Usage: Asynchronous command

CHANnel<m>:INVert <InvertChannel>

Turns the inversion of the signal amplitude on or off. To invert means to reflect the voltage values of all signal components against the ground level. If the inverted channel is the trigger source, the instrument triggers on the inverted signal.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<InvertChannel> ON | OFF
*RST: OFF

Usage: Asynchronous command

Firmware/software: FW 3.30

CHANnel<m>:BANDwidth <BandwidthLimit>

Selects the bandwidth limit for the indicated channel.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<BandwidthLimit> FULL

FULL
Use full bandwidth.

B200 | B20
Limit to 200 MHz or 20 MHz.

*RST: FULL

Usage: Asynchronous command

CHANnel<m>:CPLing <Channel>, <State>
CHANnel<m>:CPLing? <Channel>

Sets the vertical settings of the coupled channel to the values of the active channel.

Suffix:
<m> 1..4
Active channel

Parameters:
<State> OFF | ON

Parameters for setting and query:
<Channel> CHAN1 | C1 | CHAN2 | C2 | CHAN3 | C2 | CHAN4 | C4

CHAN1 = C1, CHAN2 = C2, CHAN3 = C2, CHAN4 = C4
Input channel that takes the vertical settings of CHANnel<m>.

Usage: Asynchronous command

CHANnel<m>:IMPedance <Impedance>

Sets the impedance of the channel for power calculations and measurements.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<Impedance> Range: 0.1 to 100E+3
Increment: 1
*RST: 50
Default unit: Ω

Usage: Asynchronous command

CHANnel<m>:OVERload <Overload>

Retrieves the overload status of the specified channel from the status bit. When the overload problem is solved, the command resets the status bit.

The overload status is returned asynchronously. Therefore, it is not possible to assign an overload to a specific acquisition during continuous acquisition. This assignment is only possible for a single acquisition.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<Overload> ON | OFF
Use OFF to reset the overload status bit.
*RST: OFF

Example:
```
ACQuire:COUNt 1
SINGle
CHANnel2:OVERload?
```
Queries the overload status of channel 2 after a single acquisition.
```
CHANnel2:OVERload OFF
```
Resets the overload status bit.

Usage: Asynchronous command

CHANnel<m>:FILDesign <FilterDesignTyp>

Selects the type of filter to be used.

Suffix:
<m> 1..4

Parameters:
<FilterDesignTyp> WALL | GAUSsian
`WALL`: Brick wall filter
*RST: WALL

Usage: Asynchronous command

24.8.6 Waveform data

To set the export data format, see `FORMat[:DATA]` on page 1292.

CHANnel<m>[:WAVeform<n>]:DATA:HEADer?..... 1342
CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?..... 1342

CHANnel<m>[:WAVeform<n>]:DATA:HEADer?

Returns the header of channel waveform data.

Table 24-1: Header data

Position	Meaning	Example
1	XStart in s	-9.477E-008 = - 94,77 ns
2	XStop in s	9.477E-008 = 94,77 ns
3	Record length of one waveform	200000
4	Number of values per sample interval. For most waveforms the result is 1, for peak detect and envelope waveforms it is 2. If the number is 2, the number of returned values is twice the number of samples (record length).	1

If multichannel export is enabled, the number of returned samples is *Record length * Number of exported waveforms*. See also `EXPort:WAVeform:MULTichannel`.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Example:
```
CHAN1:WAV1:DATA:HEAD?
-9.477E-008,9.477E-008,200000,1
```

Usage: Query only

CHANnel<m>[:WAVeform<n>]:DATA[:VALues]? [<Offset>], [<Length>]

Returns the data of the channel waveform points for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

Without parameters, the complete waveform is retrieved. Using the offset and length parameters, data can be retrieved in smaller portions, which makes the command faster.

To set the export format, use `FORMat[:DATA]`.

You can retrieve only Y-values (usually voltage values), or X- and Y-values. Use `EXPort:WAVeform:INCXvalues` to define which values are retrieved.

If multichannel export is active (`EXPort:WAVeform:MULTichannel`), the channel suffix is ignored. To select the channels to be exported, use `CHANnel<m>:EXPortstate`. The Y-values are written in interleaved order, for example, $YCh1_0$; $YCh2_0$; $YCh1_1$; $YCh2_1$... for a 2-channel instrument.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Query parameters:

[<Offset>] Number of offset waveform points.
Range: 0 to m. Limit: n + m <= record length

[<Length>] Number of waveform points to be retrieved.
Range: 1 to n. Limit: n + m <= record length

Return values:

<Data> List of values according to the format and content settings.

Example: Retrieve the complete channel 1 waveform, only Y-values:

```
FORM ASC
EXP:WAV:INCX OFF
CHAN1:WAV1:DATA?
<-- -0.125000,-0.123016,-0.123016,-0.123016,-0.123016,-0.123016,...
```

Example: Retrieve the first 10 values of the waveform:

```
CHANnel:WAVeform:DATA:VALues? 0,10
<-- -0.10079051554203,-0.098814234137535,-0.098814234137535,
    -0.096837945282459,-0.094861663877964,-0.094861663877964,
    -0.092885382473469,-0.090909093618393,-0.090909093618393,
    -0.088932812213898
```

Example: Skip 5 samples and retrieve the next 5 samples:

```
CHANnel:WAVeform:DATA:VALues? 5,5
<-- -0.094861663877964,-0.092885382473469,-0.090909093618393,
    -0.090909093618393,-0.088932812213898
```

Usage: Query only

24.8.7 Probes

`TRPRobe:...` command are dedicated commands for the external trigger input.

- Common probe settings....1344
- Micro button and R&S ProbeMeter....1347
- Passive probes....1350
- Active voltage probes....1352
- Modular probes....1354
- Modular probes, deembedding....1358

- Predefined probes....1361
- Current probes....1362
- Probe attributes....1363
- Probe adapter R&S RT-Z2T....1365

24.8.7.1 Common probe settings

PROBe<m>:SETup:STATe?....1344
PROBe<m>:SETup:TYPE?....1344
PROBe<m>:SETup:NAME?....1344
PROBe<m>:SETup:BANDwidth?....1345
PROBe<m>:SETup:ATTenuation[:AUTO]?....1345
PROBe<m>:SETup:OFFSet:AZERo....1345
PROBe<m>:SETup:OFFSet:USEautozero....1346
CHANnel<m>:EATScale....1346
CHANnel<m>:EATTenuation....1346

PROBe<m>:SETup:STATe?

Queries if the probe at the specified input channel is active (detected) or not active (not detected). To switch the probe on, use `CHANnel<m>:STATe`.

Suffix:
<m> 1..4

Return values:
<State> DETected | NDETected
*RST: NDETected

Usage: Query only
Asynchronous command

PROBe<m>:SETup:TYPE?

Queries the type of the probe.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<Type> String containing one of the following values:
– None (no probe detected)
– Passive Probe
– active single-ended

Usage: Query only
Asynchronous command

PROBe<m>:SETup:NAME?

Queries the name of the probe.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<Name> Name string

Usage: Query only
Asynchronous command

PROBe<m>:SETup:BANDwidth?

Queries the bandwidth of the probe.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<Bandwidth> Range: 10000 to 20E+9
*RST: 1E+9
Default unit: Hz

Usage: Query only
Asynchronous command

PROBe<m>:SETup:ATTenuation[:AUTO]?

Queries the attenuation of the probe.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<PrbAttMdAuto> Range: 1E-3 to 1000
*RST: 1
Default unit: V/V

Usage: Query only
Asynchronous command

PROBe<m>:SETup:OFFSet:AZERo

Performs an automatic correction of the zero error. If the DUT is ground-referenced, the AutoZero function can improve the measurement results.

See also: "Detect AutoZero, Use AutoZero" on page 147

Suffix:
<m> 1..4
Selects the input channel.

Usage: Event
Asynchronous command

PROBe<m>:SETup:OFFSet:USEautozero <AutoZeroOffset>

Includes the AutoZero offset in measurement results. The auto zero error is detected with `PROBe<m>:SETup:OFFSet:AZERo`.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<AutoZeroOffset> ON | OFF
*RST: OFF

Usage: Asynchronous command

Firmware/software: Version 2.70

CHANnel<m>:EATScale <ExtAttScl>

Sets the attenuation scale for an external divider.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<ExtAttScl> LIN | LOG
*RST: LIN

Usage: Asynchronous command

CHANnel<m>:EATTenuation <ExtAtt>

Sets the attenuation of an external voltage divider that is part of the DUT before the measuring point. The external attenuation is included in the measurement, and the instrument shows the results that would be measured before the divider.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<ExtAtt> Values and unit depend on the selected scale (`CHANnel<m>:EATScale`).
Range: Linear scale: 1E-3 to 1E+6, logarithmic scale: -60 dB to 120 dB
Increment: 0.01
*RST: 1

Usage: Asynchronous command

24.8.7.2 Micro button and R&S ProbeMeter

PROBe<m>:SETup:MODE 1347
PROBe<m>:SETup:DISPlaydiff 1348
PROBe<m>:PMETer:VISibility 1348
PROBe<m>:PMETer:RESults:SINGle? 1349
PROBe<m>:PMETer:RESults:COMMon? 1349
PROBe<m>:PMETer:RESults:DIFFerential? 1349
PROBe<m>:PMETer:RESults:NEGative? 1350
PROBe<m>:PMETer:RESults:POSitive? 1350

PROBe<m>:SETup:MODE <Mode>

Select the action that is started when you press the micro button on the probe head.

See also: "Micro button action" on page 148.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<Mode> RCONtinuous | RSINgle | AUToset | AZERo | SEToffsettomean | SITFile | NOACtion | FINDtriglevel | REPort | PROBemode | PRSetup

RCONtinuous
Run continuous: The acquisition is running as long as the probe button is pressed.

RSINgle
Run single: Starts a defined number of acquisitions (same as [Single] key).

AUTOSET
Starts the autoset procedure.

AZero
AutoZero: performs an automatic correction of the zero error.

SEToffsettomean
Set offset to mean: performs an automatic compensation for a DC component of the input signal.

SITFile
Save Image To File:
Directs the display image to a file. The `MMEMory:NAME` command defines the file name. The file format is defined with `HCOPy:DEVice<m>:LANGuage`.

NOACtion
Nothing is started on pressing the micro button.

FINDtriglevel
Sets the trigger level automatically to *0.5 * (MaxPeak – MinPeak)*. The function is not available for an external trigger source.

REPort
Creates and saves a report of the current results.

PROBemode
Only available for R&S RT-ZM probes. Changes the measurement mode of the probe.

PRSetup
Opens the "Probes Setup" dialog box.

*RST: RCONtinuous

Usage: Asynchronous command

PROBe<m>:SETup:DISPlaydiff <DisplayDiff>

Selects the input voltages to be measured by the ProbeMeter of an R&S differential active probe.

See also: "Differential active probes" on page 118.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<DisplayDiff> DIFFerential | SINGleended

DIFFerential
Measures differential and common mode voltages

SINGleended
Measures the voltage between the positive/negative signal socket and the ground.

*RST: DIFFerential

Usage: Asynchronous command

PROBe<m>:PMETer:VISibility <Visibility>

Activates the integrated R&S ProbeMeter of active R&S probes.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<Visibility> ON | OFF

*RST: OFF

Usage: Asynchronous command

Firmware/software: FW 2.25

PROBe<m>:PMETer:RESults:SINGle?

Returns the ProbeMeter measurement result of single-ended active R&S probes, the voltage measured between the probe tip and the ground.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<Result> Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Query only
Asynchronous command

Firmware/software: FW 2.25

PROBe<m>:PMETer:RESults:COMMon?

Returns the ProbeMeter measurement result of differential active R&S probes: the common mode voltage, which is the mean voltage between the signal sockets and the ground socket.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<Result> Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Query only
Asynchronous command

Firmware/software: FW 2.25

PROBe<m>:PMETer:RESults:DIFFerential?

Returns the ProbeMeter measurement result of differential active R&S probes, the differential voltage - the voltage between the positive and negative signal sockets.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<DiffMeasRes> Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Query only
Asynchronous command

Firmware/software: FW 2.25

PROBe<m>:PMETer:RESults:NEGative?

Returns the ProbeMeter measurement result of differential active R&S probes, the voltage that is measured between the negative signal socket and the ground.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<SgEndNegMeasRes> Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Query only
Asynchronous command

Firmware/software: FW 2.25

PROBe<m>:PMETer:RESults:POSitive?

Returns the ProbeMeter measurement result of differential active R&S probes, the voltage that is measured between the negative signal socket and the ground.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<Result> Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Query only
Asynchronous command

Firmware/software: FW 2.25

24.8.7.3 Passive probes

PROBe<m>:SETup:ATTenuation:MODE....1351
PROBe<m>:SETup:ATTenuation:UNIT....1351
PROBe<m>:SETup:ATTenuation:MANual....1351
PROBe<m>:SETup:GAIN:MANual....1351

PROBe<m>:SETup:ATTenuation:MODE <PrbAttMd>

Set the mode to MANual if the instrument does not detect the probe.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<PrbAttMd> AUTO | MANual
*RST: AUTO

Usage: Asynchronous command

PROBe<m>:SETup:ATTenuation:UNIT <AttenuationUnit>

Sets the unit for the connected probe type if `PROBe<m>:SETup:ATTenuation:MODE` on page 1351 is set to `MANual`.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<AttenuationUnit> V | A | W
Voltage probe (V), current probe (A), power probe (W)
*RST: V

Usage: Asynchronous command

PROBe<m>:SETup:ATTenuation:MANual <ManualAttenuation>

Sets the attenuation for the connected probe if `PROBe<m>:SETup:ATTenuation:MODE` on page 1351 is set to `MANual`.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<ManualAttenuation> Range: 100E-6 to 10000
Increment: 0.1
*RST: 1
Default unit: Depends on the selected unit

Usage: Asynchronous command

PROBe<m>:SETup:GAIN:MANual <GainManual>

Sets the gain of a current probe.

Suffix:
<m> 1..4
Selects the input channel. The number of channels depends on the instrument.

Parameters:
<GainManual> Range: 100E-6 to 10000
Increment: 100E-6
*RST: 1
Default unit: V/A

Usage: Asynchronous command

24.8.7.4 Active voltage probes

PROBe<m>:SETup:CMOFfset 1352
PROBe<m>:SETup:ZAXV 1352
PROBe<m>:SETup:ACCoupling 1353
PROBe<m>:SETup:ADVanced:PMToffset 1353
PROBe<m>:SETup:ADVanced:RANGe 1353
PROBe<m>:SETup:ADVanced:FILTer 1354
PROBe<m>:SETup:ADVanced:AUDioverload 1354

PROBe<m>:SETup:CMOFfset <CMOffset>

Sets the common-mode offset. The setting is only available for differential probes.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<CMOffset> Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

PROBe<m>:SETup:ZAXV <ExtAttRTZA15>

If you use the external attenuator R&S RT-ZA15 together with one of the differential active probes R&S RT-ZD10/20/30, enable it to include the external attenuation in the measurements.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<ExtAttRTZA15> ON | OFF
*RST: OFF

Usage: Asynchronous command

PROBe<m>:SETup:ACCoupling <ProbeCouplingAC>

Enables AC coupling in the R&S RT-ZPR20 probe, which removes DC and very low-frequency components. The R&S RT-ZPR20 probe requires 50 Ω input termination, for which the channel AC coupling is not available. The probe setting allows AC coupling also at 50 Ω inputs.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<ProbeCouplingAC> ON | OFF
*RST: OFF

Usage: Asynchronous command

PROBe<m>:SETup:ADVanced:PMToffset

Sets the measured ProbeMeter value as offset. Make sure that the ProbeMeter is active before you use this command.

Only available for power rail probes R&S RT-ZPR.

Suffix:
<m> 1..4
Selects the input channel.

Usage: Event
Asynchronous command

PROBe<m>:SETup:ADVanced:RANGe <ProbeRange>

Sets the voltage range of a R&S RT-ZHD probe.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<ProbeRange> AUTO | MHIGh | MLOW

AUTO
The voltage range is set with `CHANnel<m>:SCALe`.

MHIGh
Sets the higher voltage range of the connected probe. To query the value, use `PROBe<m>:SETup:ATTenuation[:AUTO]?`.

MLOW
Sets the lower voltage range of the connected probe. To query the value, use `PROBe<m>:SETup:ATTenuation[:AUTO]?`.

*RST: AUTO

Usage: Asynchronous command

PROBe<m>:SETup:ADVanced:FILTer <BwFilterSt>

Enables the lowpass filter in the probe control box.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<BwFilterSt> ON | OFF
*RST: OFF

Usage: Asynchronous command

PROBe<m>:SETup:ADVanced:AUDioverload <Sound>

Activates the acustic overrange warning in the probe control box.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<Sound> ON | OFF
*RST: OFF

Usage: Asynchronous command

24.8.7.5 Modular probes

The commands of this chapter are relevant for R&S RT-ZM modular probes. The commands are available in firmware version 3.40 and higher.

The suffix <m> selects the input channel to which the probe is connected.

PROBe<m>:SETup:PRMode....1354
PROBe<m>:SETup:DMOFfset....1355
PROBe<m>:SETup:CMOFfset....1355
PROBe<m>:SETup:NOFFset....1356
PROBe<m>:SETup:POFFset....1356
PROBe<m>:SETup:TERM:STATe....1356
PROBe<m>:SETup:TERM:MODE....1357
PROBe<m>:SETup:TERM:MEASure?....1357
PROBe<m>:SETup:TERM:ADJust....1357

PROBe<m>:SETup:PRMode <MeasMode>

Sets the measurement mode of the modular probe.

Suffix:
<m> 1..4

Parameters:
<MeasMode> DMODe | CMODe | PMODe | NMODe

DMODe
Differential mode input voltage (V_{dm}), the voltage between the positive and negative input terminal.

CMODe
Common mode input voltage (V_{cm}) , the mean voltage between the positive and negative input terminal vs. ground.

PMODe
Positive single-ended input voltage (V_p). the voltage between the positive input terminal and ground.

NMODe
Negative single-ended input voltage (V_N). the voltage between the negative input terminal and ground.

*RST: DMODe

Usage: Asynchronous command

PROBe<m>:SETup:DMOFfset <DMOffset>

Sets the differential offset to compensate a DC voltage applied to the positive and the negative input terminal.

Suffix:
<m> 1..4

Parameters:
<DMOffset> In "DM" probe mode (`DMODe`), this offset is used as channel offset and considered automatically for correction.

Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

PROBe<m>:SETup:CMOFfset <CMOffset>

Sets the common-mode offset. The setting is only available for differential probes.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:

<CMOffset> Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

PROBe<m>:SETup:NOFFset <NOffset>

Sets the negative offset to compensate a DC voltage applied to the negative input terminal (Vp) referenced to ground.

Suffix:

<m> 1..4

Parameters:

<NOffset> In "N" probe mode (`NMODe`), this offset is used as channel offset and considered automatically for correction.

Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

PROBe<m>:SETup:POFFset <POffset>

Sets the positive offset to compensate a DC voltage applied to the positive input terminal (Vp) referenced to ground.

Suffix:

<m> 1..4

Parameters:

<POffset> In "P" probe mode (`PMODe`), this offset is used as channel offset and considered automatically for correction.

Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

PROBe<m>:SETup:TERM:STATe <VoltageState>

Activates the instrument control of the termination voltage.

Suffix:

<m> 1..4

Parameters:
<VoltageState> ON | OFF
*RST: OFF

Usage: Asynchronous command

PROBe<m>:SETup:TERM:MODE <Mode>

Setting the termination voltage is relevant if you use the R&S RT-ZMA40 SMA module.

You can set a termination voltage to correct the internal 50 Ω termination of the SMA module by the common mode voltage. To control the termination voltage by the instrument, connect the VT terminal of the R&S RT-ZM probe amplifier to the VT terminal of the SMA module using the red DC lead (see R&S RT-ZM User Manual).

Suffix:
<m> 1..4

Parameters:
<Mode> AUTO | MANual

AUTO
The instrument uses the measured common mode voltage to control the termination.

MANual
Enter the voltage to be used for termination.

*RST: AUTO

Usage: Asynchronous command

PROBe<m>:SETup:TERM:MEASure?

Returns the measured common mode voltage.

Suffix:
<m> 1..4

Return values:
<VoltageMeas> Common mode voltage

Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Query only
Asynchronous command

PROBe<m>:SETup:TERM:ADJust <VoltageAdjust>

Sets the voltage to be used for termination correction.

Suffix:
<m> 1..4

Parameters:

<VoltageAdjust> Correction voltage

Range: -100E+24 to 100E+24
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

24.8.7.6 Modular probes, deembedding

The commands of this chapter are relevant for R&S RT-ZM modular probes when used for measurements on signal frequencies of 4 GHz and higher. The commands are available in firmware version 4.00 and higher.

See also: PROBe<m>:SETup:PRMode on page 1354

PROBe<m>:DEEMbedding:PMODe....1358
PROBe<m>:DEEMbedding:TIPModule[:SELect]....1358
PROBe<m>:DEEMbedding:TIPModule:CSPFile....1359
PROBe<m>:DEEMbedding:TIPModule:ZMA<n>:SUBModule....1359
PROBe<m>:DEEMbedding:BANDwidth....1359
PROBe<m>:DEEMbedding:MODE....1360
PROBe<m>:DEEMbedding:REMProbeload....1360
PROBe<m>:DEEMbedding:LOAD:IMPedance....1361

PROBe<m>:DEEMbedding:PMODe <MeasMode>

Suffix:

<m> 1..4

Parameters:

<MeasMode> DMODe | CMODe | PMODe | NMODe

See PROBe<m>:SETup:PRMode on page 1354.

*RST: DMODe

Usage: Asynchronous command

PROBe<m>:DEEMbedding:TIPModule[:SELect] <ProbeTip>

Selects the tip module that is used for measurement.

Suffix:

<m> 1..4
Selects the channel to which the probe is connected.

Parameters:

<ProbeTip> NONE | ZMA10 | ZMA12 | ZMA14 | ZMA15 | ZMA30 | ZMA40 | ZMA50 | CUST

*RST: ZMA10

Usage: Asynchronous command

PROBe<m>:DEEMbedding:TIPModule:CSPFile <SParamFile>

Sets the file name, file format and path of the S-parameter file, and loads the S-parameters of user-defined tip modules

Suffix:
<m> 1..4

Parameters:
<SParamFile> String with path, file name, and extension.

Usage: Asynchronous command

PROBe<m>:DEEMbedding:TIPModule:ZMA<n>:SUBModule <Value>

Defines additional settings for some tip modules:

For the browser module R&S RT-ZMA30, measure the space between the pins, and select the appropriate value.

If R&S RT-ZMA40 is selected, choose the used submodule: semi-rigid cables or none.

Suffix:
<m> 1..4
Selects the channel to which the probe is connected.

<n> 30 (R&S RT-ZMA30) or 40 (R&S RT-ZMA40)

Parameters:
<Value> D16 | D25 | D45 | D75 | NONE | SRCS

For R&S RT-ZMA30:
D16 = 1.6 mm
D25 = 2.5 mm
D45 = 4.5 mm
D75 = 7.5 mm
For R&S RT-ZMA40:
NONE = no submodule
SRCS = semi-rigid cables

Usage: Asynchronous command

PROBe<m>:DEEMbedding:BANDwidth <ProbeBandwidth>

Sets the maximum bandwidth until which the signal is corrected. This maximum value is the minimum bandwidth value of probe, probe tip and oscilloscope bandwidth, and can not be higher than the highest frequency in a used S-parameter file. Consider that most tip modules support the full bandwidth of the probe amplifier, but some tip modules have limited bandwidth.

Suffix:
<m> 1..4
Selects the channel to which the probe is connected.

Parameters:

<ProbeBandwidth> Range: 10000 to 16E+9
Increment: 10
*RST: 4E+9
Default unit: Hz

Usage: Asynchronous command

PROBe<m>:DEEMbedding:MODE <Mode>

Sets the probing mode.

Suffix:

<m> 1..4
Selects the channel to which the probe is connected.

Parameters:

<Mode> PROBing | TERMinating

PROBing
For measurements with a high-ohmic probe and measure on an existing line in parallel to the load.

TERMinating
For measurements if the measuring equipment is the load of the line.

*RST: PROBing

Usage: Asynchronous command

PROBe<m>:DEEMbedding:REMProbeload <RemoveProbeLoad>

Removes the loading of the probe if `DEEMbedding<m>:COMPonent<n>:MODE` is set to `PROBing`.

If the probe loading is removed, you see the signal that would be at the measurement point if the probe's input impedance is ideal (infinite impedance).

Without removing the loading, you see the real signal at the measurement point, including the probe loading.

Suffix:

<m> 1..4
Selects the channel to which the probe is connected.

Parameters:

<RemoveProbeLoad> ON | OFF

*RST: ON

Usage: Asynchronous command

PROBe<m>:DEEMbedding:LOAD:IMPedance <SourceImpedance>

Shows the source impedance of the DUT. If an R&S RT-ZM probe is connected, the value depends on the selected probe mode: common mode, differential, or single-ended measurement.

Suffix:
<m> 1..4
Selects the channel to which the probe is connected.

Parameters:
<SourceImpedance> Range: 100E-15 to 1E+9
Increment: 1
*RST: 50
Default unit: Ω

Usage: Asynchronous command

24.8.7.7 Predefined probes

PROBe<m>:SETup:ATTenuation:DEFProbe 1361
PROBe<m>:SETup:OFFSet:TOMean 1362

PROBe<m>:SETup:ATTenuation:DEFProbe <PredefinedProbe>

Selects a predefined probe. These are probes that are not recognized automatically but the parameters of the probe are known to the instrument.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<PredefinedProbe> ZC10 | ZC20 | ZC30 | ZD01A100 | ZD01A1000 | ZZ80 | FREE | ZS10L | ZD02 | ZD08 | ZC02100 | ZC021000 | ZC03 | ZD002A10 | ZD002A100 | ZD003A20 | ZD003A200 | ZC3110 | ZC311 | ZC3101

ZC10 | ZC20 | ZC30 | ZZ80 | ZS10L | ZD02 | ZD08 | ZC03
Type of the probe

ZD01A100 | ZD01A1000 | ZD002A10 | ZD002A100 | ZD003A20 | ZD003A200
High voltage differential probes R&S RT-ZD0xx, attenuation ratio according to the setting on the probe.
A10 = 10:1
A20 = 20:1
A100 = 100:1
A200 = 200:1
A1000 = 1000:1

ZC02100 | ZC021000
Current probes 100 A or 1000 A according to the setting on the probe.

FREE
Any other probe that is not recognized by the instrument.
*RST: FREE

Usage: Asynchronous command

PROBe<m>:SETup:OFFSet:TOMean

Performs an automatic compensation for a DC component of the specified input signal using the result of a background mean measurement.

Suffix:
<m> 1..4
Selects the input channel.

Usage: Event
Asynchronous command

24.8.7.8 Current probes

To set up R&S RT-ZC10 and R&S RT-ZC20, use `PROBe<m>:SETup:ATTenuation:DEFProbe`.

PROBe<m>:SETup:DEGauss 1362
PROBe<m>:SETup:OFFSet:STPRobe 1362
PROBe<m>:SETup:OFFSet:ZADJust 1363

PROBe<m>:SETup:DEGauss

Demagnetizes the core if it has been magnetized by switching the power on and off, or by an excessive input. Always carry out demagnetizing before measurement. The demagnetizing process takes about one second.

Suffix:
<m> 1..4
Selects the input channel.

Usage: Event
Asynchronous command

Firmware/software: FW 2.50

PROBe<m>:SETup:OFFSet:STPRobe

Saves the zero adjust value in the probe box. If you connect the probe to another channel or to another R&S RTx oscilloscope, the value is read out again.

Suffix:
<m> 1..4
Selects the input channel.

Usage: Event
Asynchronous command

PROBe<m>:SETup:OFFSet:ZADJust <ZroAdjVal>

set the waveform to zero position. It corrects the effect of a voltage offset or temperature drift. To set the value by the instrument, use `PROBe<m>:SETup:OFFSet:AZERo`.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<ZroAdjVal> Range: -100 to 100
Increment: 0.1
*RST: 0
Default unit: %

Usage: Asynchronous command

Firmware/software: FW 2.50

24.8.7.9 Probe attributes

PROBe<m>:ID:SWVersion?....1363
PROBe<m>:ID:PRDate?....1363
PROBe<m>:ID:PARTnumber?....1364
PROBe<m>:ID:SRNumber?....1364
PROBe<m>:SETup:CAPacitance?....1364
PROBe<m>:SETup:IMPedance?....1364

PROBe<m>:ID:SWVersion?

Queries the version of the probe firmware.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<Softwareversion> Version number in a string.

Usage: Query only
Asynchronous command

PROBe<m>:ID:PRDate?

Queries the production date of the probe.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<ProductionDate> Date in a string.

Usage: Query only
Asynchronous command

PROBe<m>:ID:PARTnumber?

Queries the R&S part number of the probe.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<PartNumber> Part number in a string.

Usage: Query only
Asynchronous command

PROBe<m>:ID:SRNumber?

Queries the serial number of the probe.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<SerialNo> Serial number in a string.

Usage: Query only
Asynchronous command

PROBe<m>:SETup:CAPacitance?

Queries the input capacitance of the probe.

Suffix:
<m> 1..4
Selects the input channel.

Return values:
<InputCapacity> Range: 100E-15 to 1E-9
*RST: 10E-12
Default unit: F

Usage: Query only
Asynchronous command

PROBe<m>:SETup:IMPedance?

Queries the termination of the probe.

Suffix:
<m> 1..4
Selects the input channel.

Return values:

<InputImpedance> Range: 0 to 1E+9
*RST: 50
Default unit: Ω

Usage: Query only
Asynchronous command

24.8.7.10 Probe adapter R&S RT-Z2T

PROBe<m>:SETup:ADAPter?.....1365
PROBe<m>:SETup:ATTenuation:TDEFprobe.....1365

PROBe<m>:SETup:ADAPter?

Queries the adapter status, whether the instrument identified the adapter.

Suffix:

<m> 1..4
Selects the input channel.

Return values:

<Type> NONE | Z2T
*RST: NONE

Usage: Query only
Asynchronous command

PROBe<m>:SETup:ATTenuation:TDEFprobe <TekPredefProbe>

Selects the Tektronix probe that is connected to the R&S RT-Z2T adapter.

Suffix:

<m> 1..4

Parameters:

<TekPredefProbe> NONE | P5205A50 | P5205A500 | P5210A100 | P5210A1000 | P6205 | P6241 | P6243 | P6245 | P6246A1 | P6246A10 | P6247A1 | P6247A10 | P6248A1 | P6248A10 | P6249 | P6250A5 | P6250A50 | P6251A5 | P6251A50 | P6701B | P6703B | P6711 | P6713 | TCP202
*RST: NONE

Example:

```
PROBe2:SETup:ADAPter
<-- Z2T
PROBe2:SETup:ATTenuation:TDEFprobe P5205A50
```

Checks the adapter state on channel 2, and selects the probe P5205A50.

Usage: Asynchronous command

24.8.8 R&S RT-ZVC probe

- Probe setup....1366
- Resolution and data....1377

24.8.8.1 Probe setup

In the `ZVC:Z<m>:` commands, the following suffixes are used:

- `<m>` selects the multi-channel probe. The available range is `1..2`
- `<n>` selects the voltage or current channel. The available range depends on the probe characteristics and is `1..2` or `1..4`

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

ZVC:BANDwidth....1367
ZVC:TYPE....1367
ZVC:Z<m>:I<n>:BANDwidth?....1367
ZVC:Z<m>:I<n>:IMPedance....1368
ZVC:Z<m>:I<n>:OFFSet....1368
ZVC:RESCoupled....1368
ZVC:Z<m>:I<n>:OVERload:RSTO....1369
ZVC:Z<m>:I<n>:OVERload:VALue?....1369
ZVC:Z<m>:I<n>:POSition....1369
ZVC:Z<m>:I<n>:SCALe....1370
ZVC:Z<m>:I<n>:SHUNt:EVALue....1370
ZVC:Z<m>:I<n>:SHUNt:MODE....1370
ZVC:Z<m>:I<n>:SHUNt:MAXCurrent....1371
ZVC:Z<m>:I<n>:SHUNt:MXCValue?....1371
ZVC:Z<m>:I<n>:SHUNt:MAXVoltage....1372
ZVC:Z<m>:I<n>:SKEW....1372
ZVC:Z<m>:I<n>[:STATe]....1372
ZVC:Z<m>:V<n>:BANDwidth?....1373
ZVC:Z<m>:V<n>:IMPedance....1373
ZVC:Z<m>:V<n>:OFFSet....1373
ZVC:Z<m>:V<n>:OVERload:RSTO....1374
ZVC:Z<m>:V<n>:OVERload:VALue?....1374
ZVC:Z<m>:V<n>:POSition....1374
ZVC:Z<m>:V<n>:SCALe....1374
ZVC:Z<m>:V<n>:SKEW....1375

ZVC:Z<m>:V<n>[:STATe]....1375
ZVC:Z<m>:ID:NAME?....1375
ZVC:Z<m>:ID:PARTnumber?....1376
ZVC:Z<m>:ID:SRNumber?....1376
ZVC:Z<m>:ID:SWVersion?....1376
ZVC:Z<m>:I<n>:ZERComp:DETect....1376
ZVC:Z<m>:I<n>:ZERComp:USE....1377
ZVC:Z<m>:I<n>:ZERComp:STATe?....1377

ZVC:BANDwidth <Bandwidth>

Sets the bandwidth limit of the probe. The bandwidth specifies the maximum frequency at which a purely sinusoidal signal is still transferred at 89 % (0.1 dB) of its amplitude.

Parameters:

<Bandwidth>
Range: 5000 to 1E+6
Increment: 5000
*RST: 1E+6
Default unit: Hz

Usage: Asynchronous command

ZVC:TYPE <DecimationMode>

Sets the decimation mode for the R&S RT-ZVC probe. Decimation reduces the data stream of the ADC to a stream of waveform points with lower sample rate and a less precise time resolution.

Parameters:

<DecimationMode> SAMPle | PDETect | HRESolution
*RST: SAMPle

Usage: Asynchronous command

ZVC:Z<m>:I<n>:BANDwidth?

Queries the bandwidth of the current channel. You can set the probe bandwidth with `ZVC:BANDwidth`.

The bandwidth of some current channels is restricted to 300KHz due to their vertical settings.

Suffix:

<m> 1..2

<n> 1..4

Return values:

<Bandwidth>
Range: 5000 to 1E+6
Increment: 5000
*RST: 1E+6
Default unit: Hz

Usage: Query only
Asynchronous command

ZVC:Z<m>:I<n>:IMPedance <MeasImp>

Sets the current impedance of the probe channel for power calculations and measurements.

Suffix:
<m> 1..2

<n> 1..4

Parameters:
<MeasImp> Range: 1 to 100E+3
Increment: 1
*RST: 50
Default unit: Ω

Usage: Asynchronous command

ZVC:Z<m>:I<n>:OFFSet <VerticalOffset>

Sets the offset current for the current channel.

Suffix:
<m> 1..2

<n> 1..4

Parameters:
<VerticalOffset> Range: -1 to 1
Increment: 0.01
*RST: 0
Default unit: V

Usage: Asynchronous command

ZVC:RESCoupled <CoupleToAnalog>

Sets the resolution of all R&S RT-ZVC channels.

Parameters:
<CoupleToAnalog> ON | OFF

ON
The resolution of the analog channels is applied to R&S RT-ZVC channels. The signal is automatically interpolated or decimated to get the analog resolution.

OFF
The resolution of R&S RT-ZVC channels is set in a way so that the record length of the waveforms is minimum 1000 samples.

*RST: OFF

Usage: Asynchronous command

ZVC:Z<m>:I<n>:OVERload:RSTO <Settings>

Resets the overload indication at the probe.

In internal shunt mode (`ZVC:Z<m>:I<n>:SHUNt:MODE` is set to `INTShunt`), chooses whether to keep the operation range adjusted by the R&S RT-ZVC during overload or to restore the original settings for the given current channel.

In external shunt mode (`ZVC:Z<m>:I<n>:SHUNt:MODE` is set to `EXTShunt`), there is no adjustment of the operation range during overload.

Suffix:

<m> 1..2

<n> 1..4

Setting parameters:

<Settings> ADJusted | ORIGinal

ADJusted
The operation range that is automatically adjusted by the probe during overload is kept.
In internal shunt mode, during an overload at a current channel, the probe switches automatically to the next higher range. If the 10A range is also exceeded, the amperemeter switches to external shunt mode to protect the probe against permanent damage.

ORIGinal
The original operation range is restored as before the overload and the adjustment of the operation range.

Usage: Setting only
Asynchronous command

ZVC:Z<m>:I<n>:OVERload:VALue?

Queries if an overload of the current channel was detected.

Suffix:

<m> 1..2

<n> 1..4

Return values:

<Overload> ON | OFF

*RST: OFF

Usage: Query only
Asynchronous command

ZVC:Z<m>:I<n>:POSition <VertPosi>

Sets the vertical position of the indicated current channel as a graphical value.

Suffix:
<m> 1..2

<n> 1..4

Parameters:
<VertPosi> Range: -5 to 5
Increment: 0.02
*RST: 0
Default unit: div

Usage: Asynchronous command

ZVC:Z<m>:I<n>:SCALe <VerticalScale>

Sets the vertical scale for the current channel in Volts per division. Increasing the scale compresses the display of the signal.

Suffix:
<m> 1..2

<n> 1..4

Parameters:
<VerticalScale> Range: 1E-15 to 10E+24
Increment: 10E-6
*RST: 3
Default unit: V/div

Usage: Asynchronous command

ZVC:Z<m>:I<n>:SHUNt:EVALue <ExtShuntVal>

Defines the value of the external shunt resistor to calculate the correct current values.

Suffix:
<m> 1..2

<n> 1..4

Parameters:
<ExtShuntVal> Range: 10E-6 to 10000
Increment: 0.02
*RST: 1
Default unit: Ω

Usage: Asynchronous command

ZVC:Z<m>:I<n>:SHUNt:MODE <ShuntMode>

Sets the internal or external shunt mode.

Suffix:
<m> 1..2

<n> 1..4

Parameters:
<ShuntMode> INTShunt | EXTShunt
*RST: INTShunt

Usage: Asynchronous command

ZVC:Z<m>:I<n>:SHUNt:MAXCurrent <IntShuntMeasRg>

Sets the maximum current and the internal shunt value.

The internal shunt and maximum current parameter pairs are defined as described in the table below.

Parameter	Internal shunt	Maximum current
A100r01	10 A	10 mΩ
A40R01	4.5 A	10 mΩ
A45M10r	45 mA	10 Ω
A4M510r	4.5 mA	10 Ω
A45u10k	45 µA	10 KΩ
A4U510k	4.5 µA	10 KΩ

Suffix:
<m> 1..2

<n> 1..4

Parameters:
<IntShuntMeasRg> A100r01 | A40R01 | A45M10r | A4M510r | A45u10k | A4U510k
*RST: A100r01

Usage: Asynchronous command

ZVC:Z<m>:I<n>:SHUNt:MXCValue?

Queries the maximum current.

Suffix:
<m> 1..2

<n> 1..4

Return values:
<MaxCurrentValue> Default unit: A

Usage: Query only
Asynchronous command

ZVC:Z<m>:I<n>:SHUNt:MAXVoltage <ExtShuntMeasRg>

Sets the maximum voltage for the external shunt.

Suffix:
<m> 1..2

<n> 1..4

Parameters:
<ExtShuntMeasRg> V450m | V045m

`V450m`: 450 mV
`V045m`: 45 mV

*RST: V450m

Usage: Asynchronous command

ZVC:Z<m>:I<n>:SKEW <DeskewOffset>

Sets the skew offset value for the current probe channel. This is a delay value, that is known from the circuit specifics but cannot be compensated by the instrument automatically.

Suffix:
<m> 1..2

<n> 1..4

Parameters:
<DeskewOffset> Range: -6E-6 to 6E-6
Increment: 200E-9
*RST: 0
Default unit: s

Usage: Asynchronous command

ZVC:Z<m>:I<n>[:STATe] <State>

Enables the corresponding current channel of the probe.

Suffix:
<m> 1..2

<n> 1..4

Parameters:
<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

ZVC:Z<m>:V<n>:BANDwidth?

Queries the bandwidth of the voltage channel. You can set the probe bandwidth with `ZVC:BANDwidth`.

Suffix:

<m>	1..2
<n>	1..4

Return values:

<Bandwidth>	Range:	5000 to 1E+6
	Increment:	5000
	*RST:	1E+6
	Default unit:	Hz

Usage: Query only
Asynchronous command

ZVC:Z<m>:V<n>:IMPedance <MeasImp>

Sets the voltage impedance of the probe channel for power calculations and measurements.

Suffix:

<m>	1..2
<n>	1..4

Parameters:

<MeasImp>	Range:	1 to 100E+3
	Increment:	1
	*RST:	50
	Default unit:	Ω

Usage: Asynchronous command

ZVC:Z<m>:V<n>:OFFSet <VerticalOffset>

Sets the vertical offset for the voltage channel.

Suffix:

<m>	1..2
<n>	1..4

Parameters:

<VerticalOffset>	Range:	-1 to 1
	Increment:	0.01
	*RST:	0
	Default unit:	V

Usage: Asynchronous command

ZVC:Z<m>:V<n>:OVERload:RSTO

Resets the overload indication at the probe.

Suffix:

<m>	1..2
<n>	1..4

Usage: Event
Asynchronous command

ZVC:Z<m>:V<n>:OVERload:VALue?

Queries if an overload of the voltage channel was detected.

Suffix:

<m>	1..2
<n>	1..4

Return values:

<Overload> ON | OFF
*RST: OFF

Usage: Query only
Asynchronous command

ZVC:Z<m>:V<n>:POSition <VertPosi>

Sets the vertical position of the indicated voltage channel as a graphical value.

Suffix:

<m>	1..2
<n>	1..4

Parameters:

<VertPosi>
Range: -5 to 5
Increment: 0.02
*RST: 0
Default unit: div

Usage: Asynchronous command

ZVC:Z<m>:V<n>:SCALe <VerticalScale>

Sets the vertical scale for the voltage channel in Volts per division. Increasing the scale compresses the display of the signal.

Suffix:

<m>	1..2
<n>	1..4

Parameters:

<VerticalScale>	Range:	1E-15 to 10E+24
	Increment:	10E-6
	*RST:	3
	Default unit:	V/div

Usage: Asynchronous command

ZVC:Z<m>:V<n>:SKEW <DeskewOffset>

Sets the skew offset value for the voltage probe channel. This is a delay value, that is known from the circuit specifics but cannot be compensated by the instrument automatically.

Suffix:

<m>	1..2
<n>	1..4

Parameters:

<DeskewOffset>	Range:	-6E-6 to 6E-6
	Increment:	200E-9
	*RST:	0
	Default unit:	s

Usage: Asynchronous command

ZVC:Z<m>:V<n>[:STATe] <State>

Enables the corresponding voltage channel of the probe.

Suffix:

<m>	1..2
<n>	1..4

Parameters:

<State>	ON | OFF	
	*RST:	OFF

Usage: Asynchronous command

ZVC:Z<m>:ID:NAME?

Queries the name of the probe.

Suffix:

<m>	1..2

Return values:
<Name>

Usage: Query only
Asynchronous command

ZVC:Z<m>:ID:PARTnumber?

Queries the R&S part number of the probe.

Suffix:
<m> 1..2

Return values:
<PartNumber>

Usage: Query only
Asynchronous command

ZVC:Z<m>:ID:SRNumber?

Queries the serial number of the probe.

Suffix:
<m> 1..2

Return values:
<SerialNo>

Usage: Query only
Asynchronous command

ZVC:Z<m>:ID:SWVersion?

Queries the version of the probe firmware.

Suffix:
<m> 1..2

Return values:
<Softwareversion>

Usage: Query only
Asynchronous command

ZVC:Z<m>:I<n>:ZERComp:DETect

Measures the zero offset, the mean value on a currentless DUT. If temperature changes, or if you change the shunt or other probe settings, repeat the measurement.

Suffix:
<m> 1..2

<n> 1..4

Usage: Event
Asynchronous command

ZVC:Z<m>:I<n>:ZERComp:USE <UseZeroComp>

Enables the automatic compensation of the measured zero offset.

Suffix:
<m> 1..2

<n> 1..4

Parameters:
<UseZeroComp> ON | OFF
*RST: OFF

Usage: Asynchronous command

ZVC:Z<m>:I<n>:ZERComp:STATe?

Returns the status of the zero compensation of the indicated current channel.

Suffix:
<m> 1..2

<n> 1..4

Return values:
<ZeroCompState> ON | OFF
ON: Zero compensation is used
Off: Zero Compensation is not used
*RST: OFF

Usage: Query only
Asynchronous command

24.8.8.2 Resolution and data

ACQuire:POINts:ZVALue?.....1377
ACQuire:ZRESolution?.....1378
ZVC:Z<m>:I<n>:DATA:HEADer?.....1378
ZVC:Z<m>:V<n>:DATA:HEADer?.....1378
ZVC:Z<m>:I<n>:DATA[:VALues]?.....1378
ZVC:Z<m>:V<n>:DATA[:VALues]?.....1378

ACQuire:POINts:ZVALue?

Returns the current record length used by the R&S RT-ZVCmulti-channel power probe channel.

Return values:
<ZUIRecordLength> Range: 1000 to 400E6
Increment: 2
*RST: 1000
Default unit: pts

Usage: Query only
Asynchronous command

ACQuire:ZRESolution?

Returns the current resolution of the R&S RT-ZVCmulti-channel power probe channel.

Return values:

<ZUIResolution> Range: 1E-15 to 0.5
Increment: 10E-12
*RST: 500E-6
Default unit: s

Usage: Query only
Asynchronous command

ZVC:Z<m>:I<n>:DATA:HEADer?
ZVC:Z<m>:V<n>:DATA:HEADer?

Return the header of amperemeter and voltmeter waveform data.

Table 24-2: Header data

Position	Meaning	Example
1	Start time in s	-0.0001
2	Stop time in s	0.0001
3	Number of values in one waveform (record length)	1000
4	Number of values per sample interval. For most waveforms the result is 1, for peak detect and envelope waveforms it is 2. If the number is 2, the number of returned values is twice the number of values (record length)..	1

Suffix:

<m> 1..2

<n> 1..4

Example:

```
ZVC:Z2:V2:DATA:HEADer?
<-- -0.0001,0.0001,1000,1
```

Usage: Query only

ZVC:Z<m>:I<n>:DATA[:VALues]?
ZVC:Z<m>:V<n>:DATA[:VALues]?

Return the waveform data of the amperemeter and voltmeter for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

Set the export format to ASCii or REAL,32 using `FORMat[:DATA]`. Export of raw data is not supported.

Suffix:

<m> 1..2

<n> 1..4

Return values:

<Data> Liste of values according to the format setting.

Example:

```
FORMat:DATA ASCii
ZVC:Z2:V2:DATA:HEADer?
<-- -0.0001,0.0001,1000,1
ZVC:Z2:V2:DATA:VaLues?
<-- 0.00083049136446789,-0.0042279558256269,-0.0076707201078534,
-0.011294682510197,-0.015915233641863,-0.018905002623796,
-0.022423267364502, ….
```

1000 values are returned.

Usage: Query only

24.8.9 Differential signals

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

DIFFerential<m>:STATe....1380
DIFFerential<m>:PSIGnal[:SELect]....1380
DIFFerential<m>:NSIGnal[:SELect]....1380
DIFFerential<m>:AOUTput....1380
DIFFerential<m>:BOUTput....1380
DIFFerential<m>:COMMon:STATe....1381
DIFFerential<m>:DIFFerential:STATe....1381
DIFFerential<m>:NSIGnal:STATe....1381
DIFFerential<m>:PSIGnal:STATe....1381
DIFFerential<m>:COMMon:SCALe....1381
DIFFerential<m>:DIFFerential:SCALe....1381
DIFFerential<m>:COUPling....1381
DIFFerential<m>:COMMon:OFFSet....1382
DIFFerential<m>:DIFFerential:OFFSet....1382
DIFFerential<m>:COMMon:POSition....1382
DIFFerential<m>:DIFFerential:POSition....1382

DIFFerential<m>:STATe <State>

Activates the differential signal. Two differential signals are available.

Suffix:
<m> 1..2
Differential signal

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

DIFFerential<m>:PSIGnal[:SELect] <Signal>
DIFFerential<m>:NSIGnal[:SELect] <Signal>

Select the analog input channel of the positive and negative signal.

Suffix:
<m> 1..2
Differential signal

Parameters:
<Signal> C1W1 | C2W1 | C3W1 | C4W1
*RST: C1W1

Usage: Asynchronous command

DIFFerential<m>:AOUTput <Output>
DIFFerential<m>:BOUTput <Output>

Select the waveform that is the result of differential processing. 2 output waveforms per digital signal can be defined.

Suffix:
<m> 1..2
Differential signal

Parameters:
<Output> P | N | DIFF | COM

P | N
Positive or negative waveform.

DIFF
Differential waveform.

COM
Common mode waveform.

*RST: COM

Usage: Asynchronous command

DIFFerential<m>:COMMon:STATe <State>
DIFFerential<m>:DIFFerential:STATe <State>
DIFFerential<m>:NSIGnal:STATe <State>
DIFFerential<m>:PSIGnal:STATe <State>

The commands enable the output waveforms if deembedding by software is active. In this case, all output waveforms can be displayed at the same time, and each output is enabled separately.

Suffix:
<m> 1..2
Differential signal

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

DIFFerential<m>:COMMon:SCALe <Scale>
DIFFerential<m>:DIFFerential:SCALe <Scale>

Set the vertical scale of differential and common mode waveforms, respectively.

Vertical settings of P and N output are the vertical settings of the input channels.

Suffix:
<m> 1..2
Differential signal

Parameters:
<Scale> Scale value, given in Volts per division.
Range: 1E-3 to 100
Increment: 1E-3
*RST: 0.05
Default unit: V/div

Usage: Asynchronous command

DIFFerential<m>:COUPling <ScaleCoupling>

If enabled for a differential signal, the vertical scales of P, N, differential and common mode waveforms are coupled. The scales are related as follows:

$VertScale_P = VertScale_N = VertScale_{CM} = VertScale_{Diff}/2$

You can disable the scale coupling and set an individual scale for each waveform.

Suffix:
<m> 1..2
Differential signal

Parameters:
<ScaleCoupling> ON | OFF
*RST: ON

Usage: Asynchronous command

DIFFerential<m>:COMMon:OFFSet <Offset>
DIFFerential<m>:DIFFerential:OFFSet <Offset>

Set the offset of differential and common mode waveforms.

Vertical settings of P and N output are the vertical settings of the input channels.

Suffix:
<m> 1..2
Differential signal

Parameters:
<Offset> Offset value
Range: -1 to 1
Increment: 0.01
*RST: 0
Default unit: V

Usage: Asynchronous command

DIFFerential<m>:COMMon:POSition <Position>
DIFFerential<m>:DIFFerential:POSition <Position>

Set the vertical position of differential and common mode waveforms.

Vertical settings of P and N output are the vertical settings of the input channels.

Suffix:
<m> 1..2
Differential signal

Parameters:
<Position> Position value
Range: -5 to 5
Increment: 0.02
*RST: 0
Default unit: div

Usage: Asynchronous command

24.8.10 Digital filter

CHANnel<m>:DIGFilter:STATe 1383
CHANnel<m>:DIGFilter:CUToff 1383
TRIGger<m>:COUPling 1383
TRIGger<m>:RFReject 1384

CHANnel<m>:DIGFilter:STATe <State>

Enables the DSP filter for input channels.

Suffix:
<m> 1..4

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

CHANnel<m>:DIGFilter:CUToff <CutOffLP>

Sets the limit frequency of the lowpass filter for input channels.

Suffix:
<m> 1..4
Selects the input channel. One filter is applied to a pair of channels - one filter for channels 1 and 2 and another filter for channels 3 and 4 (only 4-channel models).

Parameters:
<CutOffLP> Range: 100E+3 to instrument bandwidth
*RST: 1E+6
Default unit: Hz
Available filter values are:
100E+3, 200E+3, 300E+3, 500E+3, 1.0E+6, 2.0E+6, 3.0E+6, 5.0E+6, 10E+6, 20E+6, 30E+6, 50E+6, 100E+6, 500E+6, 1.0E+9, 1.5E+9, 2.0E+9, 2.5E+9, 3.0E+9, 3.5E+9, 4.0E+9, 4.5E+9, 5.0E+9, 5.5E+9, 6.0E+9

Usage: Asynchronous command

TRIGger<m>:COUPling <DigTrigCpl>

Selects the filter for the trigger channel(s). Other channels must use the same filter, or proceed unfiltered.

Suffix:
<m> 1..3
Irrelevant, omit the suffix.

Parameters:
<DigTrigCpl> OFF | RFReject

OFF
The trigger signal is not filtered.

RFReject
Frequencies higher a given limit are rejected, lower frequencies pass the filter. The limit is set with `TRIGger<m>:RFReject`.

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:RFReject <Bandwidth>

Suffix:
<m> 1..3
Irrelevant, omit the suffix.

Parameters:
<Bandwidth> Range: 100E+3 to 2E+9
Increment: 1000
*RST: 1E+6
Default unit: Hz

Usage: Asynchronous command

24.8.11 Skew

CHANnel<m>:SKEW:MANual 1384
CHANnel<m>:SKEW:TIME 1384
PROBe<m>:SKEState 1385
CHANnel<m>:SKEW:EXTended 1385

CHANnel<m>:SKEW:MANual <ManualCompens>

If enabled, the skew offset value (`CHANnel<m>:SKEW:TIME`) is used for compensation. This improves horizontal and trigger accuracy.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<ManualCompens> ON | OFF
*RST: ON

Usage: Asynchronous command

CHANnel<m>:SKEW:TIME <Offset>

Sets an delay value, that is known from the circuit specifics but cannot be compensated by the instrument automatically. It affects only the selected input channel.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:

<Offset>
Range: -100E-9 to 100E-9
Increment: 10E-15
*RST: 0
Default unit: s

Usage: Asynchronous command

PROBe<m>:SKEState <ProbeSkew>

If enabled, the skew of all connected active probes is measured, displayed, and used for deskewing.

Suffix:

<m> 1..4
The suffix is irrelevant. The setting affects all active channels.

Parameters:

<ProbeSkew> ON | OFF
*RST: OFF

Usage: Asynchronous command

CHANnel<m>:SKEW:EXTended <ExtendSkew>

Enables a higher skew value range that can be set with `CHANnel<m>:SKEW:TIME` on page 1384. The skew extension reduces the maximum number of acquisitions in the memory.

Suffix:

<m> 1..4
The suffix is irrelevant. The setting affects all active channels.

Parameters:

<ExtendSkew> ON | OFF
*RST: OFF

Usage: Asynchronous command

24.8.12 AUX OUT

CALibration:SOURce:FREQuency 1386
CALibration:SOURce:STATe 1386

CALibration:SOURce:FREQuency <Frequency>

Frequency value and waveform type of the internal calibration source, which is output at the [Aux Out] connector.

Parameters:

<Frequency> KHZ7900 | MHZ10 | KHZ15800 | MHZ31_25 | KHZ31600 | KHZ63200 | GHZ1 | MHZ250MB | MHZ250mb | GHZ1_20DB | MHZ62 | MHZ125 | MHZ250 | MHZ500 | KHZ505500 | MHZ1R | MHZ31R | MHZ31_25R | MHZ498S | MHZ1257_2S | MHZ2491S | MHZ4060S | MHZ4350S | MHZ4810S | MHZ4991S | MHZ5940S | MHZ10 | MHZ31_25 | MHZ250MB | MHZ250mb | GHZ1_20DB | GHZ1

MHZ10
10 MHz square wave

MHZ31_25 | GHZ1 | MHZ250MB
31.25 MHz, 1 GHz, 250 MHz sine waves generated by the mainboard.
MHZ250MB = MHZ250mb

GHZ1_20DB
Clinched sine wave with indicated frequency, generated by the frontend.

GHZ1
1 GHz sine wave generated by the mainboard.

*RST: GHZ1

Example:

```
CAL:SOUR:FREQ GHZ1
CAL:SOUR:STAT ON
```

The commands activate the 1 GHz reference frequency.

Usage: Asynchronous command

CALibration:SOURce:STATe <State>

Defines the state of the internal calibration source.

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

24.8.13 High definition mode

HDEFinition:STATe....1387
HDEFinition:BWIDth....1387
HDEFinition:RESolution?....1387
FORMat:BORDer....1387

HDEFinition:STATe <State>

Activates the high definition mode of the instrument.

Parameters:

<State> ON | OFF

ON: high definition mode, up to 16 bit digital resolution
OFF: normal oscilloscope mode

*RST: OFF

Usage: Asynchronous command

HDEFinition:BWIDth <Bandwidth>

Sets the filter bandwidth for the high definition mode.

Parameters:

<Bandwidth>
Range: 10000 to max. 2 GHz, depending on the instrument bandwidth.
Increment: 1000
*RST: 1E+6
Default unit: Hz

Usage: Asynchronous command

See "Bandwidth" on page 134 for bandwidth limits.

HDEFinition:RESolution?

Returns the resulting digital resolution in high definition mode.

Return values:

<Resolution>
Range: 0 to 16
Increment: 0.1
*RST: 0
Default unit: bit

Usage: Query only
Asynchronous command

FORMat:BORDer <ByteOrder>

Sets the endianess.

The command is only relevant for raw data export in high definition mode (16 bit word length).

Parameters:

<ByteOrder> LSBFirst | MSBFirst

LSB first: little endian, least significant byte first
MSB first: big endian, most significant byte first

*RST: LSBFirst

Usage: Asynchronous command

24.8.14 Reference clock

Commands are related to the external reference input on the rear panel, and the output of the OCXO reference signal.

SENSe[:ROSCillator]:SOURce 1388
SENSe[:ROSCillator]:EXTernal:FREQuency 1388

SENSe[:ROSCillator]:SOURce <RefSource>

Enables the use of the external reference signal instead of the internal OCXO reference.

Parameters:
<RefSource> INTernal | EXTernal
*RST: INTernal

Usage: Asynchronous command

SENSe[:ROSCillator]:EXTernal:FREQuency <ExternalRef>

Sets the frequency of an external reference input signal that is connected to the external reference input on the rear panel.

Parameters:
<ExternalRef> *RST: 10E+6
Default unit: Hz

Usage: Asynchronous command

24.9 Trigger

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

- Basic trigger settings 1389
- Edge trigger 1392
- Glitch trigger 1393

- Width trigger....1394
- Runt trigger....1396
- Window trigger....1398
- Timeout trigger....1400
- Interval trigger....1401
- Slew rate trigger....1403
- Setup & Hold....1405
- State trigger....1407
- Pattern trigger....1408
- Serial pattern trigger....1410
- TV/Video trigger....1412
- Trigger qualification....1417
- Holdoff....1420
- Conditioning....1424
- Trigger sequence....1426
- Trigger control....1429
- Actions on trigger....1431
- Zone trigger....1433
- External trigger input....1434
- Acquisition info....1437

24.9.1 Basic trigger settings

TRIGger<m>:SOURce[:SELect]....1389
TRIGger<m>:TYPE....1390
TRIGger<m>:LEVel<n>[:VALue]....1391
TRIGger<m>:FINDlevel....1392
DISPlay:TRIGger:LINes....1392

TRIGger<m>:SOURce[:SELect] <SourceDetailed>

Selects the source of the trigger signal.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger
Available values depend on the selected trigger source. For input channels CHAN1...4, a trigger sequence can be configured.
For all other trigger sources, only suffix 1 is allowed.

Parameters:

<SourceDetailed> CHAN1 | CHANnel1 | CHAN2 | CHANnel2 | CHAN3 | CHANnel3 | CHAN4 | CHANnel4 | EXTernanalog | SBUS | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | LOGIC | MSOB1 | MSOB2 | MSOB3 | MSOB4 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4
Input channels

EXTernanalog
External analog signal connected to the External Trigger Input. For this source, only the analog edge trigger is available.

SBUS
Serial bus

D0...D15
Digital channels (option R&S RTO6-B1)
See also: Chapter 24.18.4, "Trigger settings for digital signals and parallel buses", on page 2431

LOGIc
Logic combination of digital channels, used as trigger source (option R&S RTO6-B1)

MSOB1 | MSOB2 | MSOB3 | MSOB4
Parallel bus (option R&S RTO6-B1)

Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4
Input channels of R&S RT-ZVCmulti-channel power probe. Only available in the A-trigger with trigger type EDGE.

DIFF1 | DIFF2 | COMMON1 | COMMON2
Differential signals

*RST: CHAN1

Usage: Asynchronous command

TRIGger<m>:TYPE <Type>

Selects the trigger type to trigger on analog channels or the external trigger input.

See also: Chapter 6.3, "Trigger types", on page 193.

To trigger on digital channels and parallel buses, use `TRIGger<m>:PARallel:TYPE`.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger
For suffixes 2 and 3, the following trigger types are availabe: EDGE, GLITch, WIDTh, RUNT, WINDow, TIMeout, INTerval, SLEWrate.

Parameters:
<Type> EDGE | GLITch | WIDTh | RUNT | WINDow | TIMeout | INTerval | SLEWrate | STATe | PATTern | ANEDge | SERPattern | TV | | SETHold | | NFC | CDR

Most of the type values are self-explanatory.

SETHold
SETHold: analyzes the relative timimg between a data signal and the synchronous clock signal. For trigger settings, see Chapter 24.9.10, "Setup & Hold", on page 1405.

ANEDge
Edge trigger for external trigger input. Only available if the trigger source is the external trigger input. This trigger type uses the analog input signal. For trigger settings, see Chapter 6.10.1, "External trigger setup", on page 236.

SERPattern
Serial pattern for signals with serial data patterns in relation to a clock signal. For trigger settings, see Chapter 24.9.13, "Serial pattern trigger", on page 1410

NFC
Specific trigger for Near Field Communication testing, requires option R&S RTO6--K11. For NFC trigger settings, see Chapter 24.21.3, "NFC trigger", on page 2478.

CDR
Trigger on the edges of a recovered clock signal, requires option R&S RTO6--K13. The CDR also uses the trigger level as threshold to detect signal edges. For CDR trigger settings, see Chapter 24.22.4, "Clock data recovery (hardware-based, option R&S RTO6-K13)", on page 2515.

*RST: EDGE

Usage: Asynchronous command

TRIGger<m>:LEVel<n>[:VALue] <Level>

Sets the trigger level for the specified event and source.

If the trigger source is serial bus, the trigger level is set by the thresholds in the protocol configuration.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

<n> 1..31
Indicates the trigger source:
1...4 = channel 1 to 4
5 = external trigger input
6...11 = not available
12...27 = R&S RT-ZVC input channels
28...31 = differential signals Diff1, Diff2 and Common1, Common2

Parameters:

<Level> Voltage for the trigger level.

Range: Depends on vertical scale, channel offset and other settings. The trigger level must be within the current display range.
Increment: 1E-3
*RST: 0
Default unit: V

Example: `TRIG:LEV5 0.01`
Sets the trigger level for the external trigger signal to 10 mV.
`TRIG2:LEV3 0.2`
Sets the trigger level for the B-event and B-trigger source channel 3 to 200 mV.

Usage: Asynchronous command

TRIGger<m>:FINDlevel

Sets the trigger level automatically. The command is only relevant if the trigger source is an analog channel CHAN1...4.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Usage: Event
Asynchronous command

DISPlay:TRIGger:LINes <State>

Hides or shows the trigger levels in the diagrams.

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

24.9.2 Edge trigger

TRIGger<m>:EDGE:SLOPe 1392

TRIGger<m>:EDGE:SLOPe <Slope>

Defines the edge for the edge trigger event.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<Slope> POSitive | NEGative | EITHer

See Chapter 24.4.3, "Slope parameter", on page 1285.

*RST: POSitive

Usage: Asynchronous command

24.9.3 Glitch trigger

TRIGger<m>:GLITch:POLarity....1393
TRIGger<m>:GLITch:RANGe....1393
TRIGger<m>:GLITch:WIDTh....1394

TRIGger<m>:GLITch:POLarity <Polarity>

Defines the polarity of a pulse, that is the direction of the first pulse slope.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<Polarity> POSitive | NEGative | EITHer

See Chapter 24.4.4, "Polarity parameter", on page 1285.

*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:GLITch:RANGe <RangeMode>

Selects which glitches are identified: shorter or longer than the width specified using `TRIGger<m>:GLITch:WIDTh`.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<RangeMode> SHORter | LONGer

SHORter
Glitches shorter than the specified width are identified.

LONGer
Glitches longer than the specified width are identified.

*RST: SHORter

Usage: Asynchronous command

TRIGger<m>:GLITch:WIDTh <Width>

Sets the length of a glitch. The instrument triggers on pulses shorter or longer than this value, depnding on the `TRIGger<m>:GLITch:RANGe` command.

You need to know the expected pulse widths of the circuit to set the glitch width correctly.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<Width> Range: 100E-12 to 10000
Increment: 100E-6
*RST: 1E-9
Default unit: s

Usage: Asynchronous command

24.9.4 Width trigger

TRIGger<m>:WIDTh:POLarity....1394
TRIGger<m>:WIDTh:RANGe....1394
TRIGger<m>:WIDTh:WIDTh....1395
TRIGger<m>:WIDTh:DELTa....1395

TRIGger<m>:WIDTh:POLarity <Polarity>

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<Polarity> POSitive | NEGative
See Chapter 24.4.4, "Polarity parameter", on page 1285.
*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:WIDTh:RANGe <RangeMode>

Defines how the range of a pulse width is defined in relation to the width and delta specified using `TRIGger<m>:WIDTh:WIDTh` and `TRIGger<m>:WIDTh:DELTa`, respectively.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<RangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers on pulses inside a given range. The range is defined by the width ±delta.

OUTSide
Triggers on pulses outside a given range. The range is defined by the width ±delta.

SHORter
Triggers on pulses shorter than the given width.

LONGer
Triggers on pulses longer than the given width.

*RST: WITHin

Usage: Asynchronous command

TRIGger<m>:WIDTh:WIDTh <Width>

For the ranges "Within" and "Outside" (defined using `TRIGger<m>:WIDTh:RANGe`), the width defines the center of a range which is defined by the limits "±Delta" (see `TRIGger<m>:WIDTh:DELTa` on page 1395).

For the ranges "Shorter" and "Longer", the width defines the maximum and minimum pulse width, respectively.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<Width>
Range: 100E-12 to 10000
Increment: 100E-9
*RST: 5E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:WIDTh:DELTa <WidthDelta>

Defines a range around the width value specified using `TRIGger<m>:WIDTh:WIDTh`.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<WidthDelta>
Range: 0 to 432
Increment: 500E-12
*RST: 0
Default unit: s

Usage: Asynchronous command

24.9.5 Runt trigger

TRIGger<m>:RUNT:POLarity 1396
TRIGger<m>:LEVel<n>:RUNT:LOWer 1396
TRIGger<m>:LEVel<n>:RUNT:UPPer 1396
TRIGger<m>:RUNT:RANGe 1396
TRIGger<m>:RUNT:WIDTh 1397
TRIGger<m>:RUNT:DELTa 1397

TRIGger<m>:RUNT:POLarity <Polarity>

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<Polarity> POSitive | NEGative | EITHer

See Chapter 24.4.4, "Polarity parameter", on page 1285.

*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:LEVel<n>:RUNT:LOWer <Level>
TRIGger<m>:LEVel<n>:RUNT:UPPer <Level>

Set the lower and upper voltage thresholds.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

<n> 1..31
Indicates the trigger source:
1...4 = channel 1...4
5...27 = not available
28...31 = differential signals Diff1, Diff2 and Common1, Common2

Parameters:
<Level> Range: -10 to 10
Increment: 1E-3
*RST: Lower = -0.1, upper =0.1
Default unit: V

Usage: Asynchronous command

TRIGger<m>:RUNT:RANGe <Mode>

Defines the time limit of the runt pulse in relation to the `TRIGger<m>:RUNT:WIDTh` and `TRIGger<m>:RUNT:DELTa` settings.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<Mode> ANY | LONGer | SHORter | WITHin | OUTSide

ANY
Triggers on all runts fulfilling the level condition, without time limitation.

LONGer
Triggers on runts longer than the given "Runt width".

SHORter
Triggers on runts shorter than the given "Runt width".

WITHin
Triggers if the runt length is inside a given time range. The range is defined by "Runt width" and "±Delta".

OUTSide
Triggers if the runt length is outside a given time range. The range is defined by "Runt width" and "±Delta".

*RST: ANY

Usage: Asynchronous command

TRIGger<m>:RUNT:WIDTh <Width>

Defines the upper or lower voltage threshold. This command is not available if `TRIGger<m>:RUNT:RANGe` is set to "ANY".

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<Width>
Range: 100E-12 to 10000
Increment: 100E-9
*RST: 5E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:RUNT:DELTa <WidthDelta>

Defines a range around the runt width specified using `TRIGger<m>:RUNT:WIDTh`. This command is only available if `TRIGger<m>:RUNT:RANGe` is set to "WITHin" or "OUTSide".

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<WidthDelta>

Range: 100E-12 to 864
Increment: 100E-9
*RST: 100E-12
Default unit: s

Usage: Asynchronous command

24.9.6 Window trigger

TRIGger<m>:LEVel<n>:WINDow:LOWer....1398
TRIGger<m>:LEVel<n>:WINDow:UPPer....1398
TRIGger<m>:WINDow:RANGe....1398
TRIGger<m>:WINDow:TIME....1399
TRIGger<m>:WINDow:WIDTh....1400
TRIGger<m>:WINDow:DELTa....1400

TRIGger<m>:LEVel<n>:WINDow:LOWer <Level>
TRIGger<m>:LEVel<n>:WINDow:UPPer <Level>

Set the lower and upper voltage limits for the window.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

<n> 1..31
Indicates the trigger source:
1...4 = channel 1...4
5...27 = not available
28...31 = differential signals Diff1, Diff2 and Common1, Common2

Parameters:

<Level>

Range: -10 to 10
Increment: 1E-3
*RST: Lower = -0.1, upper =0.1
Default unit: V

Usage: Asynchronous command

TRIGger<m>:WINDow:RANGe <RangeMode>

Defines the signal run in relation to the window:

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<RangeMode> ENTer | EXIT | WITHin | OUTSide

ENTer
Triggers when the signal crosses the upper or lower level and thus enters the window made up of these two levels.

EXIT
Triggers when the signal leaves the window.

WITHin
Triggers if the signal stays between the upper and lower level for a specified time. The time is defined using the `TRIGger<m>:WINDow:TIME` command.

OUTSide
Triggers if the signal stays above the upper level or below the lower level for a specified time. The time is defined using the `TRIGger<m>:WINDow:TIME` command.

*RST: ENTer

Usage: Asynchronous command

TRIGger<m>:WINDow:TIME <TimeRangeMode>

Defines the limit of the window in relation to the time specified using `TRIGger<m>:WINDow:WIDTh` and `TRIGger<m>:WINDow:DELTa`. Time conditioning is available for `TRIGger<m>:WINDow:RANGe`= "WITHin" and "OUTSide".

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<TimeRangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers if the signal stays inside or outside the vertical window limits at least for the time *Width - Delta* and for *Width + Delta* at the most.

OUTSide
"Outside" is the opposite definition of "Within". The instrument triggers if the signal stays inside or outside the vertical window limits for a time shorter than *Width - Delta* or longer than *Width + Delta*.

SHORter
Triggers if the signal crosses vertical limits before the specified "Width" time is reached.

LONGer
Triggers if the signal crosses vertical limits before the specified "Width" time is reached.

*RST: WITHin

Usage: Asynchronous command

TRIGger<m>:WINDow:WIDTh <Width>

For the ranges "Within" and "Outside" (defined using `TRIGger<m>:WINDow:RANGe`), the width defines the center of a time range which is defined by the limits "±Delta" (see `TRIGger<m>:WINDow:DELTa` on page 1400).

For the ranges "Shorter" and "Longer", it defines the maximum and minimum time lapse, respectively.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<Width> Range: 100E-12 to 10000
Increment: 100E-9
*RST: 5E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:WINDow:DELTa <WidthDelta>

Defines a range around the "Width" value specified using `TRIGger<m>:WINDow:WIDTh`.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<WidthDelta> Range: 0 to 432
Increment: 500E-12
*RST: 0
Default unit: s

Usage: Asynchronous command

24.9.7 Timeout trigger

TRIGger<m>:TIMeout:RANGe....1400
TRIGger<m>:TIMeout:TIME....1401

TRIGger<m>:TIMeout:RANGe <TimeoutMode>

Defines the relation of the signal level to the trigger level.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<TimeoutMode> HIGH | LOW | EITHer

HIGH
The signal level stays above the trigger level.

LOW
The signal level stays below the trigger level.

EITHer
The signal level stays above or below the trigger level.

*RST: HIGH

Usage: Asynchronous command

TRIGger<m>:TIMeout:TIME <Time>

Defines the time limit for the timeout at which the instrument triggers.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<Time> Range: 100E-12 to 10000
Increment: 100E-9
*RST: 100E-9
Default unit: s

Usage: Asynchronous command

24.9.8 Interval trigger

TRIGger<m>:INTerval:SLOPe 1401
TRIGger<m>:INTerval:RANGe 1402
TRIGger<m>:INTerval:WIDTh 1402
TRIGger<m>:INTerval:DELTa 1402

TRIGger<m>:INTerval:SLOPe <Slope>

Sets the edge for the trigger. You can analyze the interval between positive edges or between negative edges.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<Slope> POSitive | NEGative
See Chapter 24.4.3, "Slope parameter", on page 1285.
*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:INTerval:RANGe <RangeMode>

Defines the range of an interval in relation to the interval width specified using `TRIGger<m>:INTerval:WIDTh` and `TRIGger<m>:INTerval:DELTa`.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<RangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers on pulses inside a given range. The range is defined by the interval width ±delta.

OUTSide
Triggers on pulses outside a given range. The range is defined by the interval width ±delta.

SHORter
Triggers on pulses shorter than the given interval width.

LONGer
Triggers on pulses longer than the given interval width.

*RST: OUTSide

Usage: Asynchronous command

TRIGger<m>:INTerval:WIDTh <Width>

Defines the time between two pulses.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<Width>
Range: 100E-12 to 10000
Increment: 100E-9
*RST: 5E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:INTerval:DELTa <WidthDelta>

Defines a range around the "Interval width" value specified using `TRIGger<m>:INTerval:WIDTh` on page 1402.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<WidthDelta> Range: 0 to 10
Increment: 100E-9
*RST: 0
Default unit: s

Usage: Asynchronous command

24.9.9 Slew rate trigger

TRIGger<m>:SLEW:SLOPe 1403
TRIGger<m>:LEVel<n>:SLEW:LOWer 1403
TRIGger<m>:LEVel<n>:SLEW:UPPer 1403
TRIGger<m>:SLEW:RANGe 1404
TRIGger<m>:SLEW:RATE 1404
TRIGger<m>:SLEW:DELTa 1405

TRIGger<m>:SLEW:SLOPe <Slope>

Selects the edge type for the trigger event.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<Slope> POSitive | NEGative | EITHer
See Chapter 24.4.3, "Slope parameter", on page 1285.
*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:LEVel<n>:SLEW:LOWer <Level>
TRIGger<m>:LEVel<n>:SLEW:UPPer <Level>

Set the lower and upper voltage thresholds. When the signal crosses a level, the slew rate measurement starts or stops depending on the selected slope.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

<n> 1..31
Indicates the trigger source:
1...4 = channel 1...4
5...27 = not available
28...31 = differential signals Diff1, Diff2 and Common1, Common2

Parameters:

<Level>
Range: -10 to 10
Increment: 1E-3
*RST: Lower = -0.1, upper =0.1
Default unit: V

Usage: Asynchronous command

TRIGger<m>:SLEW:RANGe <RangeMode>

Defines the time limit for the slew rate in relation to the upper or lower trigger level (see `TRIGger<m>:SLEW:RATE` on page 1404 and `TRIGger<m>:SLEW:DELTa` on page 1405). The time measurement starts when the signal crosses the first trigger level - the upper or lower level depending on the selected slope - and stops when the signal crosses the second level.

Suffix:

<m>
1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<RangeMode>
INSRange | OUTRange | LTHan | GTHan

INSRange
Triggers on pulses inside a given range. The range is defined by the slew rate ±delta.

OUTRange
Triggers on pulses outside a given range. The range is defined by the slew rate ±delta.

LTHan
Triggers on pulses shorter than the given slew rate.

GTHan
Triggers on pulses longer than the given slew rate.

*RST: GTHan

Usage: Asynchronous command

TRIGger<m>:SLEW:RATE <Time>

For the ranges "Within" and "Outside", the slew rate defines the center of a range which is defined by the limits "±Delta".

For the ranges "Shorter" and "Longer", the slew rate defines the maximum and minimum slew rate limits, respectively. When the signal crosses this level, the slew rate measurement starts or stops depending on the selected slope (see `TRIGger<m>:SLEW:SLOPe` on page 1403).

Suffix:

<m>
1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<Time>
Range: 100E-12 to 10000
Increment: 100E-9
*RST: 100E-12
Default unit: s

Usage: Asynchronous command

TRIGger<m>:SLEW:DELTa <TimeDelta>

Defines a time range around the slew rate specified using `TRIGger<m>:SLEW:RATE`.

Suffix:

<m>
1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:

<TimeDelta>
Range: 0 to 10
Increment: 100E-9
*RST: 0
Default unit: s

Usage: Asynchronous command

24.9.10 Setup & Hold

The Setup & Hold trigger is only available for the A-event (Suffix = 1).

TRIGger<m>:SETHold:CSOurce:EDGE 1405
TRIGger<m>:SETHold:CSOurce:LEVel 1406
TRIGger<m>:SETHold:CSOurce[:VALue] 1406
TRIGger<m>:SETHold:HTIMe 1406
TRIGger<m>:SETHold:STIMe 1407
TRIGger<m>:SCOupling 1407

TRIGger<m>:SETHold:CSOurce:EDGE <ClockEdge>

Sets the edge of the clock signal to define the time reference point for the setup and hold time.

Suffix:

<m>
1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<ClockEdge>
POSitive | NEGative | EITHer
See Chapter 24.4.3, "Slope parameter", on page 1285.
*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:SETHold:CSOurce:LEVel <ClockLevel>

Sets the voltage level for the clock signal. Both this command and `TRIGger<m>:SETHold:CSOurce:EDGE` define the starting point for calculation of the setup and hold time.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<ClockLevel> Range: -10 to 10
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

TRIGger<m>:SETHold:CSOurce[:VALue] <ClockSource>

Selects the source of the clock signal.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<ClockSource> CHAN1 | CHANnel1 | CHAN2 | CHANnel2 | CHAN3 | CHANnel3 | CHAN4 | CHANnel4

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4
Input channel of the clock signal

*RST: CHAN1

Usage: Asynchronous command

TRIGger<m>:SETHold:HTIMe <HoldTime>

Sets the minimum time **after** the clock edge while the data signal must stay steady above or below the data level.

The hold time can be negative. In this case, the setup time is always positive. The setup/hold interval starts before the clock edge (setup time) and ends before the clock edge (hold time). If you change the negative hold time, the setup time is adjusted by the intrument.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<HoldTime> Range: -99.999E-9 to 100E-9
Increment: 1E-9
*RST: 0
Default unit: s

Usage: Asynchronous command

TRIGger<m>:SETHold:STIMe <SetupTime>

Sets the minimum time **before** the clock edge while the data signal must stay steady above or below the data level.

The setup time can be negative. In this case, the hold time is always positive. The setup/hold interval starts after the clock edge (setup time) and ends after the clock edge (hold time). If you change the negative setup time, the hold time is adjusted by the instrument.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<SetupTime> Range: -99.999E-9 to 100E-9
Increment: 1E-9
*RST: 0
Default unit: s

Usage: Asynchronous command

TRIGger<m>:SCOupling <TrigLevSrcCpl>

Sets the trigger levels of all used channels to the level of the data line (Setup & Hold and serial pattern trigger), clock line (state trigger) or source line (pattern trigger).

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<TrigLevSrcCpl> ON | OFF
*RST: OFF

Usage: Asynchronous command

24.9.11 State trigger

The state trigger combines the edge trigger settings with trigger qualification. It is only available for the A-event (Suffix = 1).

Use the following commands:

- `TRIGger<m>:EDGE:SLOPe` on page 1392

- TRIGger<m>:LEVel<n>[:VALue] on page 1391
- TRIGger<m>:SCOupling on page 1407
- To define the state, use the TRIG:QUAL...[ENABle] and TRIG:QUAL...LOGic commands, which are described in Chapter 24.9.15, "Trigger qualification", on page 1417.

24.9.12 Pattern trigger

The pattern trigger is only available for the A-event (Suffix = 1).

The pattern is defined using the commands:

- TRIGger<m>:QUALify<n>:A[:ENABle] on page 1418
- TRIGger<m>:QUALify<n>:A:LOGic on page 1419
- TRIGger<m>:QUALify<n>:AB:LOGic on page 1420

These are the commands for channel 1, use the similar commands for channels 2, 3, and 4.

TRIGger<m>:PATTern:MODE....1408
TRIGger<m>:PATTern:TIMeout:MODE....1409
TRIGger<m>:PATTern:TIMeout[:TIME]....1409
TRIGger<m>:PATTern:WIDTh:RANGe....1409
TRIGger<m>:PATTern:WIDTh[:WIDTh]....1410
TRIGger<m>:PATTern:WIDTh:DELTa....1410

TRIGger<m>:PATTern:MODE <Mode>

Adds additional time limitation to the pattern definition.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<Mode> OFF | TIMeout | WIDTh

OFF
No time limitation. The event occurs if the pattern condition is fulfilled.

TIMeout
Defines how long the result of the pattern condition must be true or false. The duration of the timeout is defined using TRIGger<m>:PATTern:TIMeout[:TIME].

WIDTh
Defines a time range for keeping up the true result of the pattern condition. The range is defined using TRIGger<m>:PATTern:WIDTh:RANGe.

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:PATTern:TIMeout:MODE <TimeoutMode>

Defines the condition for the timeout.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<TimeoutMode> HIGH | LOW

HIGH
The result stays high.

LOW
The result stays low.

*RST: HIGH

Usage: Asynchronous command

TRIGger<m>:PATTern:TIMeout[:TIME] <Time>

Defines how long the result of the pattern condition must be true or false.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Time> Range: 100E-12 to 10000
Increment: 100E-9
*RST: 100E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:PATTern:WIDTh:RANGe <WidthRangeMode>

Defines how the range of a pulse width is defined for keeping up the true result of the pattern condition. The width and delta are specified using `TRIGger<m>:PATTern:WIDTh[:WIDTh]` and `TRIGger<m>:PATTern:WIDTh:DELTa`, respectively.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<WidthRangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers on pulses inside a given range. The range is defined by the width ±delta.

OUTSide
Triggers on pulses outside a given range. The range is defined by the width ±delta.

SHORter
Triggers on pulses shorter than the given width.

LONGer
Triggers on pulses longer than the given width.

*RST: WITHin

Usage: Asynchronous command

TRIGger<m>:PATTern:WIDTh[:WIDTh] <Width>

For the ranges "Within" and "Outside" (defined using `TRIGger<m>:PATTern:WIDTh:RANGe`), the width defines the center of a range which is defined by the limits "±Delta" (see `TRIGger<m>:PATTern:WIDTh:DELTa` on page 1410).

For the ranges "Shorter" and "Longer", the width defines the maximum and minimum pulse width, respectively.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Width>
Range: 100E-12 to 10000
Increment: 100E-9
*RST: 5E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:PATTern:WIDTh:DELTa <WidthDelta>

Defines a range around the width value specified using `TRIGger<m>:PATTern:WIDTh[:WIDTh]`.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<WidthDelta>
Range: 0 to 432
Increment: 500E-12
*RST: 0
Default unit: s

Usage: Asynchronous command

24.9.13 Serial pattern trigger

The serial pattern trigger is only available for the A-event (Suffix = 1).

TRIGger<m>:SPATtern:CSOurce[:VALue]....1411
TRIGger<m>:SPATtern:CSOurce:EDGE....1411
TRIGger<m>:SPATtern:CSOurce:LEVel....1411
TRIGger<m>:SPATtern:PATTern....1412
TRIGger<m>:SPATtern:REFSource....1412

TRIGger<m>:SPATtern:CSOurce[:VALue] <ClockSource>

Defines the source of the clock signal.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<ClockSource> CHAN1 | CHANNEL1 | CHAN2 | CHANNEL2 | CHAN3 | CHANNEL3 | CHAN4 | CHANNEL4

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4
Input channel of the clock signal

*RST: CHAN1

Usage: Asynchronous command

TRIGger<m>:SPATtern:CSOurce:EDGE <ClockEdge>

Together with the clock level (see `TRIGger<m>:SPATtern:CSOurce:LEVel` on page 1411), the clock edge defines the point in time when the state of the data signal is checked.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<ClockEdge> POSitive | NEGative | EITHer

See Chapter 24.4.3, "Slope parameter", on page 1285.

*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:SPATtern:CSOurce:LEVel <ClockLevel>

Defines the voltage level for the clock signal.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<ClockLevel> Range: -10 to 10
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

TRIGger<m>:SPATtern:PATTern <Pattern>

The pattern contains the bits of the serial data to be found in the data stream. The maximum length of the pattern is 128 bit.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<Pattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Usage: Asynchronous command

TRIGger<m>:SPATtern:REFSource <ReferenceSource>

Selects if an explicit clock or a CDR signal is used as a reference signal.

Suffix:

<m> 1..3

Parameters:

<ReferenceSource> CLOCk | CDR
*RST: CDR

Usage: Asynchronous command

24.9.14 TV/Video trigger

The TV or video trigger requires a channel input as trigger source (`TRIGger<m>:SOURce[:SELect]`). It is only available for the A-event (Suffix = 1).

Make sure to set the trigger level - the threshold of the sync pulse - with `TRIGger<m>:LEVel<n>[:VALue]`.

TRIGger<m>:TV:STANdard....1413
TRIGger<m>:TV:MODE....1413
TRIGger<m>:TV:POLarity....1414
TRIGger<m>:TV:LINE....1414
TRIGger<m>:TV:LFIeld....1415
TRIGger<m>:TV:CUSTom:SCANmode....1415

TRIGger<m>:TV:CUSTom:LDURation....1416
TRIGger<m>:TV:CUSTom:STYPe.... 1416
TRIGger<m>:TV:CUSTom:SDURation.... 1417

TRIGger<m>:TV:STANdard <Standard>

Sets the TV standard.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Standard> CUSTom | PAL | PALM | NTSC | SECam | P480L60HZ | P720L30HZ | P720L50HZ | P720L60HZ | I1080L50HZ | I1080L60HZ | P1080L24HZ | P1080L24HZSF | P1080L25HZ | P1080L30HZ | P1080L50HZ | P1080L60HZ

CUSTom
User-defined signal. Configure the signal using:
`TRIGger<m>:TV:CUSTom:SCANmode`
`TRIGger<m>:TV:CUSTom:STYPe`
`TRIGger<m>:TV:CUSTom:LDURation`
`TRIGger<m>:TV:CUSTom:SDURation`

PAL | PALM | NTSC | SECam
SDTV standards. PALM = PAL-M

PxxxxLyyHZ
HDTV standards using progressive scanning (P). xxxx indicates the number of active lines, yy is the frame rate.

IxxxxLxxHZ
HDTV standards using interlaced scanning (I). xxxx indicates the number of active lines, yy is the field rate.

P1080L24HZSF
1080p/24sF is a HDTV standard using progressive segmented frame scanning.

*RST: PAL

Usage: Asynchronous command

TRIGger<m>:TV:MODE <Mode>

Selects the lines or fields on which the instrument can trigger. Available modes depend on the scanning system.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Mode> ALL | ODD | EVEN | ALINe | LINE

ALL
All fields, triggers on the frame start (progressive scanning) or field start (interlaced and progressive segmented frame scanning)

ODD | EVEN
Only available for interlaced scanning and progressive segmented frame scanning. Triggers on the field start of the odd or even field.

ALINe
All lines, triggers on all line starts.

LINE
Triggers on a specified line. To set the line number, use `TRIGger<m>:TV:LINE`. For NTSC signals, set also the field with `TRIGger<m>:TV:LFIeld`.

*RST: ALL

Usage: Asynchronous command

TRIGger<m>:TV:POLarity <Polarity>

Sets the polarity of the *signal*. Note that the sync pulse has the opposite polarity, for example, a positive signal has a negative sync pulse.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Polarity> POSitive | NEGative
*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:TV:LINE <LineNumber>

Specifies the line number to trigger on. The command is relevant if `TRIGger<m>:TV:MODE` is set to `LINe`.

Usually the lines of the frame are counted beginning from the frame start. For NTSC signals, the lines are counted per field, not per frame. For these signals, set also the field with `TRIGger<m>:TV:LFIeld`.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<LineNumber> Range: Depends on the standard, see table below
Increment: 1
*RST: 1

Usage: Asynchronous command

Standard	Minimum value	Maximum value
PAL	1	625
PAL-M	1	525
NTSC	1	263 in odd field 262 in even field
SECAM	1	625
480p/60 (`P480L60HZ`)	1	525
720p/30 (`P720L30HZ`) 720p/50 (`P720L50HZ`) 720p/60 (`P720L60HZ`)	1	750
1080i/50 (`I1080L50HZ`) 1080i/60 (`I1080L60HZ`) 1080p/24 (`P1080L24HZ`) 1080p/24sF (`P1080L24HZSF`) 1080p/25 (`P1080L25HZ`) 1080p/30 (`P1080L30HZ`) 1080p/50 (`P1080L50HZ`) 1080p/60 (`P1080L60HZ`)	1	1125

TRIGger<m>:TV:LFIeld <LineField>

The commands is only relevant for NTSC signals and sets the field in which the line number is counted.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<LineField> FIELD1 | FIELD2
FIELD1 = odd field
FIELD2 = even field
*RST: FIELD1

Usage: Asynchronous command

TRIGger<m>:TV:CUSTom:SCANmode <ScanMode>

Sets the scanning system. Only relevant if `TRIGger<m>:TV:STANdard` is set to `CUSTom`.

See also: "Scan" on page 219.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<ScanMode> INTerlaced | PROGressive | SEGMented

SEGMented: Progressive segmented frame uses progressive scanning to capture the frame, and interlaced scanning for transmission and display.

*RST: INTerlaced

Usage: Asynchronous command

TRIGger<m>:TV:CUSTom:LDURation <LinePeriod>

Sets the duration of a line, the time between two successive sync pulses. Only relevant if `TRIGger<m>:TV:STANdard` is set to `CUSTom`.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<LinePeriod> Range: 1E-6 to 500E-6
Increment: 100E-9
*RST: 64E-6
Default unit: s

Usage: Asynchronous command

TRIGger<m>:TV:CUSTom:STYPe <SyncPulseType>

Sets the type of the sync pulse. Only relevant if `TRIGger<m>:TV:STANdard` is set to `CUSTom`.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<SyncPulseType> BILevel | TRILevel

BILevel
Bi-level sync pulse, usually used in SDTV signals

TRILevel
Tri-level sync pulse, used in HDTV signals

*RST: BILevel

Usage: Asynchronous command

TRIGger<m>:TV:CUSTom:SDURation <SyncPulseDurat>

Sets the width of the sync pulse. Only relevant if `TRIGger<m>:TV:STANdard` is set to `CUSTom`.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<SyncPulseDurat> Range: 100E-9 to 100E-6
Increment: 100E-9
*RST: 4.7E-6
Default unit: s

Usage: Asynchronous command

24.9.15 Trigger qualification

The A-event can have its own trigger qualification. Qualification is not available for B- and R-events (event suffix m = 2 | 3) and some trigger types.

A suffix defines the trigger type to which the qualification belongs.

Table 24-3: Trigger type suffixes

Suffix	Trigger type
1	EDGE
2	GLITch
3	WIDTh
4	RUNT
5	WINDow
6	TIMeout
7	INTerval
8	Qualification is not supported (SLEWrate)
9	Qualification is not supported (SETHold)
10	STATe
11	PATTern
12	qualification is not supported (ANEDge, ext. trigger input)
13	Currently not used
14	Currently not used
15	Qualification is not supported (SERPattern)
16	Qualification is not supported (NFC)

Suffix	Trigger type
17	Qualification is not supported (TV)
18	Qualification is not supported (CDR)

TRIGger<m>:QUALify<n>:STATe....1418
TRIGger<m>:QUALify<n>:A[:ENABle]....1418
TRIGger<m>:QUALify<n>:B[:ENABle]....1418
TRIGger<m>:QUALify<n>:C[:ENABle]....1418
TRIGger<m>:QUALify<n>:D[:ENABle]....1418
TRIGger<m>:QUALify<n>:A:LOGic....1419
TRIGger<m>:QUALify<n>:B:LOGic....1419
TRIGger<m>:QUALify<n>:C:LOGic....1419
TRIGger<m>:QUALify<n>:D:LOGic....1419
TRIGger<m>:QUALify<n>:AB:LOGic....1420
TRIGger<m>:QUALify<n>:CD:LOGic....1420
TRIGger<m>:QUALify<n>:ABCD:LOGic....1420

TRIGger<m>:QUALify<n>:STATe <AddTrigLogi>

Enables the use of the qualification definition for the selected trigger event.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> 1..18
Defines the trigger type, see Table 24-3.

Parameters:

<AddTrigLogi> ON | OFF

ON
The qualification expression is considered for the trigger event.

OFF
The qualification expression is ignored for the trigger event.

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:QUALify<n>:A[:ENABle] <State>
TRIGger<m>:QUALify<n>:B[:ENABle] <State>
TRIGger<m>:QUALify<n>:C[:ENABle] <State>
TRIGger<m>:QUALify<n>:D[:ENABle] <State>

The command is relevant for qualification, for pattern trigger and for state trigger.

Select the channels to be considered:

- A[:ENABle]: C1
- B[:ENABle]: C2
- C[:ENABle]: C3

- D[:ENABle]: C4

The trigger source cannot be enabled.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> 1..18
Trigger type, see Table 24-3.

Parameters:

<State> ON | OFF

ON
The qualification expression is considered.

OFF
The qualification expression is ignored.

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:QUALify<n>:A:LOGic <Operator>
TRIGger<m>:QUALify<n>:B:LOGic <Operator>
TRIGger<m>:QUALify<n>:C:LOGic <Operator>
TRIGger<m>:QUALify<n>:D:LOGic <Operator>

Defines the logic for the indicated channel:

- A: C1
- B: C2
- C: C3
- D: C4

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> 1..18
Trigger type, see Table 24-3.

Parameters:

<Operator> DIRect | HIGH | NOT | LOW

DIRect
Input value remains unchanged

NOT
Input value is inverted

*RST: DIRect

Usage: Asynchronous command

TRIGger<m>:QUALify<n>:AB:LOGic <Operator>
TRIGger<m>:QUALify<n>:CD:LOGic <Operator>
TRIGger<m>:QUALify<n>:ABCD:LOGic <Operator>

Defines the logical combination of the indicated channels after evaluating the previous logical operations:

- AB: CH1 and CH2
- CD: CH3 and CH4
- ABCD: result of AB and CD

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> 1..18
Trigger type, see Table 24-3.

Parameters:

<Operator> AND | NAND | OR | NOR

AND
logical AND, conjunctive combination

NAND
logical NOT AND

OR
logical OR, disjunctive combination

NOR
logical NOT OR

*RST: AND

Usage: Asynchronous command

24.9.16 Holdoff

TRIGger<m>:HOLDoff:MODE 1420
TRIGger<m>:HOLDoff:TIME 1421
TRIGger<m>:HOLDoff:EVENts 1422
TRIGger<m>:HOLDoff:MIN 1422
TRIGger<m>:HOLDoff:MAX 1423
TRIGger<m>:HOLDoff:AUTotime? 1423
TRIGger<m>:HOLDoff:SCALing 1423

TRIGger<m>:HOLDoff:MODE <Mode>

Selects the method to define the holdoff condition.

The trigger holdoff defines when the next trigger after the current will be recognized. Thus, it affects the next trigger to occur after the current one. Holdoff helps to obtain stable triggering when the oscilloscope is triggering on undesired events.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Mode> TIME | EVENts | RANDom | AUTO | OFF

TIME
Defines the holdoff directly as a time period. The next trigger occurs only after the "Holdoff time" has passed (defined using `TRIGger<m>:HOLDoff:TIME`).

EVENts
Defines the holdoff as a number of trigger events. The next trigger occurs only when this number of events is reached. The number of triggers to be skipped is defined using `TRIGger<m>:HOLDoff:EVENts`.

RANDom
Defines the holdoff as a random time limited by `TRIGger<m>:HOLDoff:MIN` on page 1422 and `TRIGger<m>:HOLDoff:MAX` on page 1423. For each acquisition cycle, the instrument selects a new random holdoff time from the specified range.

AUTO
The holdoff time is calculated automatically based on the current horizontal scale.

OFF
No holdoff

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:HOLDoff:TIME <Time>

Defines the holdoff time period. The next trigger occurs only after this time has passed. The setting is relevant if the holdoff mode is set to TIME.

See also:

- `TRIGger<m>:HOLDoff:MODE`

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Time> Range: 100E-9 to 10
Increment: 200E-6
*RST: 1E-3
Default unit: s

Example:
```
TRIGger:HOLDoff:MODE TIME
TRIGger:HOLDoff:TIME 1ms
```
The holdoff time is set to 1 ms.

Usage: Asynchronous command

TRIGger<m>:HOLDoff:EVENts <Events>

Defines the number of triggers to be skipped. The next trigger only occurs when this number of events is reached. The setting is relevant if the holdoff mode is set to EVENts.

See also:

- `TRIGger<m>:HOLDoff:MODE`

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Events> Range: 1 to 2147483647
Increment: 1
*RST: 1

Example:
```
TRIGger:HOLDoff:MODE EVENts
TRIGger:HOLDoff:EVENts 5
```

Usage: Asynchronous command

TRIGger<m>:HOLDoff:MIN <RandomMinTime>

Defines the lower limit for the random time holdoff. The setting is relevant if the holdoff mode is set to RANDom.

See also:

- `TRIGger<m>:HOLDoff:MODE`
- `TRIGger<m>:HOLDoff:MAX`

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<RandomMinTime> Range: 100E-9 to 5
Increment: 200E-6
*RST: 1E-3
Default unit: s

Example:
```
TRIGger:HOLDoff:MODE RANDom
TRIGger:HOLDoff:MIN 1ms
TRIGger:HOLDoff:MAX 2ms
```
The holdoff time is set randomly between 1 ms and 2 ms.

Usage: Asynchronous command

TRIGger<m>:HOLDoff:MAX <RandomMaxTime>

Defines the upper limit for the random time holdoff. The setting is relevant if the holdoff mode is set to RANDom.

See also:

- `TRIGger<m>:HOLDoff:MODE`
- `TRIGger<m>:HOLDoff:MIN`

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<RandomMaxTime> Range: 100E-9 to 10
Increment: 200E-6
*RST: 2E-3
Default unit: s

Usage: Asynchronous command

TRIGger<m>:HOLDoff:AUTotime?

Returns the resulting holdoff time if the holdoff mode is set to AUTO: *Auto time = Auto time scaling * Horizontal scale*. The auto time scaling factor is defined with `TRIGger<m>:HOLDoff:SCALing`.

See also: `TRIGger<m>:HOLDoff:MODE`

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Return values:

<AutoTime> Holdoff time

Range: 100E-9 to 10
*RST: 1E-3
Default unit: s

Example:

```
TRIGger:HOLDoff:MODE AUTO
TRIGger:HOLDoff:SCALing 0.5
TRIGger:HOLDoff:AUTotime?
1ms
```

Result if the horizontal scale is 1 ns/div

Usage: Query only
Asynchronous command

TRIGger<m>:HOLDoff:SCALing <AutoTimeScaling>

Sets the auto time scaling factor the horizontal scale is multipied with: *Auto time = Auto time scaling * Horizontal scale*. The setting is relevant if the holdoff mode is set to AUTO.

See also:

- TRIGger<m>:HOLDoff:MODE
- TRIGger<m>:HOLDoff:AUTotime? on page 1423

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:

<AutoTimeScaling> Range: 1E-3 to 1000
Increment: 1
*RST: 0.5

Usage: Asynchronous command

24.9.17 Conditioning

TRIGger<m>:LEVel<n>:NOISe[:STATe] 1424
TRIGger<m>:LEVel<n>:NOISe:MODE 1425
TRIGger<m>:LEVel<n>:NOISe:ABSolute 1425
TRIGger<m>:LEVel<n>:NOISe:PERDivision 1426
TRIGger<m>:LEVel<n>:NOISe:RELative 1426

TRIGger<m>:LEVel<n>:NOISe[:STATe] <HysteresisMode>

Selects how the hysteresis is set.

Suffix:

<m> 1..3
Irrelevant, omit the suffix.

<n> 1..31
Indicates the trigger source:
1...4 = channel 1 to 4
5 = external trigger input
6...11 = not available
12...27 = R&S RT-ZVC input channels
28...31 = differential signals Diff1, Diff2 and Common1, Common2

Parameters:

<HysteresisMode> AUTO | MANual

AUTO
This is the recommended mode. The hysteresis is set by the instrument to reject at least the internal noise of the instrument. You can define a higher minimum value using TRIGger<m>:LEVel<n>:NOISe:ABSolute.

MANual
The hysteresis is defined directly with TRIGger<m>:LEVel<n>:NOISe:ABSolute.

*RST: AUTO

Usage: Asynchronous command

TRIGger<m>:LEVel<n>:NOISe:MODE <HystMode>

Selects how the hysteresis is set.

Suffix:

<m> 1..3
Irrelevant, omit the suffix.

<n> 1..31
Indicates the trigger source: see `TRIGger<m>:LEVel<n>:NOISe[:STATe]` on page 1424.

Parameters:

<HystMode> ABS | REL

ABS
The hysteresis is set in absolute values (voltage).

REL
The hysteresis is defined in relative values (div).

*RST: ABS

Usage: Asynchronous command

TRIGger<m>:LEVel<n>:NOISe:ABSolute <HystAbs>

Defines a range in absolute values around the trigger level. If the signal oscillates inside this range and crosses the trigger level thereby, no trigger event occurs.

Suffix:

<m> 1..3
Irrelevant, omit the suffix.

<n> 1..31
Indicates the trigger source: see `TRIGger<m>:LEVel<n>:NOISe[:STATe]` on page 1424.

Parameters:

<HystAbs> Range: 0 to The value corresponding to full division range. The exact maximum value depends on the selected vertical scale.
Increment: 1E-3
*RST: 0
Default unit: V

Usage: Asynchronous command

TRIGger<m>:LEVel<n>:NOISe:PERDivision <HystInDivs>

Defines a range in divisions around the trigger level. If the signal oscillates inside this range and crosses the trigger level thereby, no trigger event occurs.

Suffix:

<m> 1..3
Irrelevant, omit the suffix.

<n> 1..31
Indicates the trigger source: see `TRIGger<m>:LEVel<n>:NOISe[:STATe]` on page 1424.

Parameters:

<HystInDivs> Hysteresis size in divisions

Range: 0 to 5
Increment: 0.01
*RST: 0
Default unit: div

Usage: Asynchronous command

TRIGger<m>:LEVel<n>:NOISe:RELative <HystRel>

Defines a range in percent around the trigger level. If the signal oscillates inside this range and crosses the trigger level thereby, no trigger event occurs.

Suffix:

<m> 1..3
Irrelevant, omit the suffix.

<n> 1..31
Indicates the trigger source: see `TRIGger<m>:LEVel<n>:NOISe[:STATe]` on page 1424.

Parameters:

<HystRel> Hysteresis in %. 10% = 1 div

Range: 0 to 50
Increment: 1
*RST: 0
Default unit: %

Usage: Asynchronous command

24.9.18 Trigger sequence

TRIGger<m>:EVENt[:EVENt] 1427
TRIGger<m>:SEQuence:TYPE 1427
TRIGger<m>:ECOupling 1428
TRIGger<m>:SEQuence:DELay 1428
TRIGger<m>:SEQuence:COUNt 1428

TRIGger<m>:SEQuence:RESet:EVENt....1428
TRIGger<m>:SEQuence:RESet:TIMeout[:ENABle]....1429
TRIGger<m>:SEQuence:RESet:TIMeout:TIME....1429

TRIGger<m>:EVENt[:EVENt] <Class>

Selects, if you want to trigger on a single event, or on a series of events.

Suffix:
<m> 1..3

Parameters:
<Class> SINGle | SEQuence
*RST: SINGle

Usage: Asynchronous command

TRIGger<m>:SEQuence:TYPE <Sequence>

Selects the type of the sequence. All trigger sequences require that input channels CH1...4 are set as trigger sources for all conditions.

See also: Chapter 6.8, "Sequence", on page 229.

Suffix:
<m> 1..3
Irrelevant, omit the suffix.

Parameters:
<Sequence> ABR | AZ | ABRZ | AORB | AORBZ

ABR
Triggers if all conditions of A- and B-events, as well as additional delay and reset timeout or R-event (reset) conditions are fulfilled.

AZ
Triggers if the conditions of the A-event and the zone trigger are fulfilled. Additionally, a holdoff condition can be set.

ABRZ
Triggers if all conditions of A- and B-events, reset timeout or R-event (reset) conditions are fulfilled for the defined zone.

AORB
Triggers if the conditions of A-event or the conditions of the B-event are fulfilled.

AORBZ
Triggers if the conditions of A-event or the conditions of the B-event are fulfilled for the defined zone. Additionally, a holdoff condition can be set.

*RST: ABR

Usage: Asynchronous command

TRIGger<m>:ECOupling <TrigLevEvtCpl>

Event coupling of trigger levels: Sets the trigger levels to the values of the indicated event. Thus, channel 1 has one trigger level for all events, channel 2 has one trigger level and so on.

Suffix:
<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

Parameters:
<TrigLevEvtCpl> ON | OFF
*RST: ON

Usage: Asynchronous command

TRIGger<m>:SEQuence:DELay <Delay>

Sets the time the instrument waits after an A-event until it recognizes B-events.

Suffix:
<m> 1..3
Irrelevant, omit the suffix.

Parameters:
<Delay> Range: 0 to 50
Increment: 1E-12
*RST: 0
Default unit: s

Usage: Asynchronous command

TRIGger<m>:SEQuence:COUNt <Events>

Sets the number of B-events to be fulfilled after an A-event. The last B-event causes the trigger.

Suffix:
<m> 1..3
Irrelevant, omit the suffix.

Parameters:
<Events> Range: 1 to 2147483647
Increment: 1
*RST: 1

Usage: Asynchronous command

TRIGger<m>:SEQuence:RESet:EVENt <EnabRstEvt>

If set to ON, the trigger sequence is restarted by the R-event if the specified number of B-event does not occur.

Suffix:
<m> 1..3
Irrelevant, omit the suffix.

Parameters:
<EnabRstEvt> ON | OFF
*RST: OFF

Usage: Asynchronous command

TRIGger<m>:SEQuence:RESet:TIMeout[:ENABle] <State>

If set to ON, the instrument waits for the time defined using `TRIGger<m>:SEQuence:RESet:TIMeout:TIME` for the specified number of B-events. If no trigger occurs during that time, the sequence is restarted with the A-event.

Suffix:
<m> 1..3
Irrelevant, omit the suffix.

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

TRIGger<m>:SEQuence:RESet:TIMeout:TIME <ResetTimeout>

The time the instrument waits for the number of B-events specified using `TRIGger<m>:SEQuence:COUNt` before the sequence is restarted with the A-event.

Suffix:
<m> 1..3
Irrelevant, omit the suffix.

Parameters:
<ResetTimeout> Range: 0 to 50
Increment: 1E-12
*RST: 0
Default unit: s

Usage: Asynchronous command

24.9.19 Trigger control

TRIGger<m>:MODE....1430
TRIGger<m>:FORCe....1430
TRIGger<m>:OUT:POLarity....1430
TRIGger<m>:OUT:PLENgth....1431
TRIGger<m>:OUT:DELay....1431

TRIGger<m>:MODE <TriggerMode>

Sets the trigger mode which determines the behaviour of the instrument if no trigger occurs.

Suffix:
<m> 1..3
Irrelevant, omit the suffix.

Parameters:
<TriggerMode> AUTO | NORMal | FREerun

AUTO
The instrument triggers repeatedly after a time interval if the trigger conditions are not fulfilled. If a real trigger occurs, it takes precedence. The time interval depends on the time base.

NORMal
The instrument acquires a waveform only if a trigger occurs.

FREerun
The instrument triggers after a very short time interval - faster than in AUTO mode. Real triggers are ignored.

*RST: AUTO

Usage: Asynchronous command

TRIGger<m>:FORCe

If the acquisition is running in normal mode and no valid trigger occurs, forcing the trigger provokes an immediate single acquisition. Thus you can confirm that a signal is available and use the waveform display to determine how to trigger on it.

Suffix:
<m> 1..3
Irrelevant, omit the suffix.

Usage: Event
Asynchronous command

TRIGger<m>:OUT:POLarity <Polarity>

Sets the polarity of the trigger out pulse.

Suffix:
<m> 1..3
Irrelevant, omit the suffix.

Parameters:
<Polarity> POSitive | NEGative

*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:OUT:PLENgth <PulseLength>

Sets the length of the trigger out pulse.

Suffix:
<m> 1..3
Irrelevant, omit the suffix.

Parameters:
<PulseLength> Range: 4E-9 to 0.06
Increment: 20E-9
*RST: 100E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:OUT:DELay <Delay>

Sets the delay of the first pulse edge to the trigger point.

The setting is not available if a mask test or measurement is running and the on-violation event is set to trigger out pulse.

Suffix:
<m> 1..3
Irrelevant, omit the suffix.

Parameters:
<Delay> Range: 800E-9 to 1
Increment: 1E-9
*RST: 800E-9
Default unit: s

Usage: Asynchronous command

24.9.20 Actions on trigger

TRIGger<m>:EVENt:BEEP 1431
TRIGger<m>:EVENt:PRINt 1432
TRIGger<m>:EVENt:WFMSave 1432
TRIGger<m>:EVENt:RUNexec 1432
TRIGger<m>:OUT:ACTion 1433

TRIGger<m>:EVENt:BEEP <Beep>

Generates a beep sound if the command is set to `TRIGger`.

Suffix:
<m> 1..3
The suffix is irrelevant.

Parameters:

<Beep> NOACtion | TRIGger

*RST: NOACtion

Usage: Asynchronous command

TRIGger<m>:EVENt:PRINt <Print>

Saves a screenshot at each trigger if the command is set to `TRIGger`.

For screenshot settings, see Chapter 24.16.8, "Screenshots", on page 1660.

Suffix:

<m> 1..3
The suffix is irrelevant.

Parameters:

<Print> NOACtion | TRIGger

*RST: NOACtion

Usage: Asynchronous command

TRIGger<m>:EVENt:WFMSave <SaveWfm>

Saves the waveform data to file at each trigger if the command is set to `TRIGger`.

For data export settings, see Chapter 24.16.4, "Waveform data export to file", on page 1649

Suffix:

<m> 1..3
The suffix is irrelevant.

Parameters:

<SaveWfm> NOACtion | TRIGger

*RST: NOACtion

Usage: Asynchronous command

TRIGger<m>:EVENt:RUNexec <RunExecutable>

Starts an external application if the command is set to `TRIGger`.

Use the following commands to set up the application:

- `EXECutable:NAME` on page 1668
- `EXECutable:PARameter` on page 1669
- `EXECutable:WDIRectory` on page 1669

Suffix:

<m> 1..3
The suffix is irrelevant.

Parameters:
<RunExecutable> NOACtion | TRIGger
*RST: NOACtion

Usage: Asynchronous command

TRIGger<m>:OUT:ACTion <StateAction>

Enables/disbales the trigger out signal that is provided to the [Trigger Out] connector on the rear panel when a trigger occurs.

Suffix:
<m> 1..3

Parameters:
<StateAction> NOACtion | TRIGger
*RST: NOACtion

Usage: Asynchronous command

24.9.21 Zone trigger

TRIGger<m>:ZONE:EXPRession[:DEFine] <LogiExpr>

Defines the zone trigger.

Suffix:
<m> 1..3
The suffix is irrelevant.

Parameters:
<LogiExpr> String with the logical expression

Example: Prerequisite: mask tests MT1 and MT2 are defined, and also Zone1.

```
TRIGger:ZONE:EXPRession 'MT1 and MT2 and Zone1'
TRIGger:ZONE:EXPRession?
<-- MT1 and MT2 and Zone1
```

Usage: Asynchronous command

Firmware/software: FW 3.20

SWTRigger:HISTory <ApplyZnTrigHistory>

Applies the zone trigger condition to the acquisitions in the history memory during history replay.

See: "Apply zone trigger to history" on page 235.

Parameters:
<ApplyZnTrigHistory> ON | OFF
*RST: OFF

Usage: Asynchronous command

24.9.22 External trigger input

External trigger signals, which are connected to the external trigger input, can be triggered with an edge trigger.

TRIGger<m>:ANEDge:COUPling <Coupling>

Sets the coupling for the external trigger signal.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Coupling> DC | DCLimit | AC

DC
Direct connection with 50 Ω termination, passes both DC and AC components of the trigger signal.

DCLimit
Direct connection with 1 MΩ termination, passes both DC and AC components of the trigger signal.

AC
Connection through DC capacitor, removes DC and very low-frequency components.

Usage: Asynchronous command

TRIGger<m>:ANEDge:FILTer <Filter>

Sets a filter for the external trigger signal to reject high or low frequencies.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Filter> OFF | LFReject | RFReject

OFF
The trigger signal is not filtered.

LFReject
Frequencies higher than the "Cut-off" frequency are rejected, lower frequencies pass the filter.
You can adjust the "Cut-off" frequency using the `TRIGger<m>:ANEDge:CUToff:LOWPass` command, the default is 50 kHz.

RFReject
Frequencies below the "Cut-off" frequency are rejected, higher frequencies pass the filter.
You can adjust the "Cut-off" frequency using the `TRIGger<m>:ANEDge:CUToff:HIGHpass` command, the default is 50 kHz.

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:ANEDge:CUToff:HIGHpass <AnalogCutOffHP>

Frequencies below the "Cut-off" frequency are rejected, higher frequencies pass the filter.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<AnalogCutOffHP> KHZ5 | KHZ50 | MHZ50

Cut-off frequency

KHZ5
5 kHz

KHZ50
50 kHz

MHZ50
50 MHz

*RST: KHZ50

Usage: Asynchronous command

TRIGger<m>:ANEDge:CUToff:LOWPass <AnalogCutOffLP>

Frequencies higher than the "Cut-off" frequency are rejected, lower frequencies pass the filter.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<AnalogCutOffLP> KHZ5 | KHZ50 | MHZ50

KHZ5
5 kHz

KHZ50
50 kHz

MHZ50
50 MHz

*RST: KHZ50

Usage: Asynchronous command

TRIGger<m>:ANEDge:GND <Ground>

Connects the analog signal to the ground.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Ground> ON | OFF
*RST: OFF

Usage: Asynchronous command

TRIGger<m>:ANEDge:SLOPe <Slope>

Sets the edge for the trigger event.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Slope> POSitive | NEGative
See Chapter 24.4.3, "Slope parameter", on page 1285.
*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:EXTern:OVERload <Overload>

The query returns the overload status of the external trigger input.

`:TRIGger:EXTern:OVERload 0` confirms the information in the message box, it has same effect as OK.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<Overload> ON | OFF

ON | 1
Iindicates an overload of the probe

OFF | 0
In a query: no overlaod.
Ssetting: confirms the information in the message box

*RST: OFF

Usage: Asynchronous command

24.9.23 Acquisition info

ACQuire:CURRent?

Shows the current number of acquisitions that have been acquired.

Return values:

<CurrAcqCnt>	Range:	0 to 4294967295
	Increment:	1
	*RST:	0

Usage: Query only
Asynchronous command

24.10 Waveform analysis

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

- Zoom....1437
- Reference waveforms....1446
- Mathematics....1456
- History....1461
- XY-Diagram....1465

24.10.1 Zoom

LAYout:ZOOM:ADD....1438
LAYout:ZOOM:ADDCoupled....1438
LAYout:ZOOM:POSCoupling....1439
LAYout:ZOOM:HORZ:MODE....1439
LAYout:ZOOM:HORZ:ABSolute:POSition....1439
LAYout:ZOOM:HORZ:ABSolute:SPAN....1440
LAYout:ZOOM:HORZ:ABSolute:STARt....1440
LAYout:ZOOM:HORZ:ABSolute:STOP....1441
LAYout:ZOOM:HORZ:RELative:POSition....1441
LAYout:ZOOM:HORZ:RELative:SPAN....1441
LAYout:ZOOM:HORZ:RELative:STARt....1442
LAYout:ZOOM:HORZ:RELative:STOP....1442

LAYout:ZOOM:VERTical:MODE....1442
LAYout:ZOOM:VERTical:ABSolute:POSition....1443
LAYout:ZOOM:VERTical:ABSolute:SPAN....1443
LAYout:ZOOM:VERTical:ABSolute:STARt....1444
LAYout:ZOOM:VERTical:ABSolute:STOP....1444
LAYout:ZOOM:VERTical:RELative:POSition....1444
LAYout:ZOOM:VERTical:RELative:SPAN....1445
LAYout:ZOOM:VERTical:RELative:STARt....1445
LAYout:ZOOM:VERTical:RELative:STOP....1445
LAYout:ZOOM:REMove....1446

LAYout:ZOOM:ADD <NodeName>, <ParentType>, <InsertBefore>, <XStart>, <XStop>, <YStart>, <YStop>, <NewZoomName>

Adds a new zoom diagram based on the specified waveform.

For numerical values, the input of a unit is not supported for this command.

Setting parameters:

<NodeName>	String with the name of diagram to be zoomed
<ParentType>	VERTical, OFF The new zoom diagram is displayed below the original one.
<InsertBefore>	OFF Position of the zoom diagram, depending on `ParentType`
<XStart>	Defines the x-value at the beginning of the zoom area.
<XStop>	Defines the x-value at the end of the zoom area.
<YStart>	Defines the y-value at the beginning of the zoom area.
<YStop>	Defines the y-value at the end of the zoom area.
<NewZoomName>	String with the name of the new zoom diagram.
Example:	`LAYout:ZOOM:ADD 'Diagram1', VERT, OFF, -10e-9, 20e-9, -0.1, 0.05, 'MyZoom1'` Creates the zoom diagram 'MyZoom1' for 'Diagram1'.
Example:	See Chapter 24.3.1.2, "Creating zoom diagrams", on page 1261
Usage:	Setting only Asynchronous command

LAYout:ZOOM:ADDCoupled <ZoomName>, <XOffset>, <YOffset>, <NewZoomName>

Creates a new zoom diagram based on the settings of an existing zoom area for the same source.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<NewZoomName>	Defines the name of the new zoom diagram.

Setting parameters:

<ZoomName> Defines the name of the zoom diagram to be copied.

<XOffset> Defines an offset to the existing zoom area in x direction.

<YOffset> Defines an offset to the existing zoom area in y direction.

Usage: Asynchronous command

LAYout:ZOOM:POSCoupling <DiagramName>, <ZoomName>, <PositionCoupl>
LAYout:ZOOM:POSCoupling? <DiagramName>, <ZoomName>

Enables or disables the position coupling of coupled zooms. If position coupling is enabled and one zoom area is moved, the other coupled zoom areas are moved, too, and keep their distance.

Parameters:

<PositionCoupl> ON | OFF

*RST: OFF

Parameters for setting and query:

<DiagramName> String with the name of the diagram on which the zoom is based

<ZoomName> String with the name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:HORZ:MODE <DiagramName>, <ZoomName>,<Mode>
LAYout:ZOOM:HORZ:MODE? <DiagramName>, <ZoomName>

Defines whether absolute or relative values are used to specify the x-axis values. Since the zoom area refers to the active signal, relative values ensure that the zoom area remains the same.

Parameters:

<Mode> ABS | REL

Mode used to specify the x-axis values of the zoom area.

*RST: ABS

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Example: See Chapter 24.3.1.2, "Creating zoom diagrams", on page 1261

Usage: Asynchronous command

LAYout:ZOOM:HORZ:ABSolute:POSition <DiagramName>, <ZoomName>,<Position>
LAYout:ZOOM:HORZ:ABSolute:POSition? <DiagramName>, <ZoomName>

Defines the x-value of the centerpoint of the zoom area.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Position>	Range:	-100E+24 to 100E+24
	Increment:	0.01
	*RST:	0.01

Parameters for setting and query:

<DiagramName>	Name of the diagram on which the zoom area is based.
<ZoomName>	Name of the zoom diagram
Usage:	Asynchronous command

LAYout:ZOOM:HORZ:ABSolute:SPAN <DiagramName>, <ZoomName>,<Span>
LAYout:ZOOM:HORZ:ABSolute:SPAN? <DiagramName>, <ZoomName>

Defines the width of the zoom area.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Span>	Range:	0 to 100E+24
	Increment:	0.01
	*RST:	0.01

Parameters for setting and query:

<DiagramName>	Name of the diagram on which the zoom area is based.
<ZoomName>	Name of the zoom diagram
Usage:	Asynchronous command

LAYout:ZOOM:HORZ:ABSolute:STARt <DiagGroupNme>,
<ZoomDiagramName>,<Start>
LAYout:ZOOM:HORZ:ABSolute:STARt? <DiagGroupNme>, <ZoomDiagramName>

Defines the lower limit of the zoom area on the x-axis.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Start>	Range:	-100E+24 to 100E+24
	Increment:	0.01
	*RST:	0

Parameters for setting and query:

<DiagGroupNme>	Name of the diagram on which the zoom area is based.
<ZoomDiagramName>	Name of the zoom diagram
Usage:	Asynchronous command

LAYout:ZOOM:HORZ:ABSolute:STOP <DiagramName>, <ZoomName>,<Stop>
LAYout:ZOOM:HORZ:ABSolute:STOP? <DiagramName>, <ZoomName>

Defines the upper limit of the zoom area on the x-axis.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Stop>
Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:HORZ:RELative:POSition <DiagramName>, <ZoomName>,<RelPosi>
LAYout:ZOOM:HORZ:RELative:POSition? <DiagramName>, <ZoomName>

Defines the x-value of the centerpoint of the zoom area.

Parameters:

<RelPosi> Relative position of the centerpoint (x-value)

Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Example: See Chapter 24.3.1.2, "Creating zoom diagrams", on page 1261

Usage: Asynchronous command

LAYout:ZOOM:HORZ:RELative:SPAN <DiagramName>, <ZoomName>,<RelativeSpan>
LAYout:ZOOM:HORZ:RELative:SPAN? <DiagramName>, <ZoomName>

Defines the width of the zoom area.

Parameters:

<RelativeSpan>
Range: 1E-15 to 100
Increment: 0.1
*RST: 1
Default unit: %

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Example: See Chapter 24.3.1.2, "Creating zoom diagrams", on page 1261

Usage: Asynchronous command

LAYout:ZOOM:HORZ:RELative:STARt <DiagramName>,
<ZoomName>,<RelativeStart>
LAYout:ZOOM:HORZ:RELative:STARt? <DiagramName>, <ZoomName>

Defines the lower limit of the zoom area on the x-axis.

Parameters:

<RelativeStart>
Range: 0 to 100
Increment: 0.1
*RST: 0
Default unit: %

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:HORZ:RELative:STOP <DiagramName>,
<ZoomName>,<RelativeStop>
LAYout:ZOOM:HORZ:RELative:STOP? <DiagramName>, <ZoomName>

Defines the upper limit of the zoom area on the x-axis.

Parameters:

<RelativeStop>
Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:MODE <DiagramName>, <ZoomName>,<Mode>
LAYout:ZOOM:VERTical:MODE? <DiagramName>, <ZoomName>

Defines whether absolute or relative values are used to specify the y-axis values. Since the zoom area refers to the active signal, relative values ensure that the zoom area remains the same.

Parameters:

<Mode> ABS | REL

Mode used to specify the y-axis values of the zoom area.

*RST: ABS

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:ABSolute:POSition <DiagramName>, <ZoomName>,<Position>
LAYout:ZOOM:VERTical:ABSolute:POSition? <DiagramName>, <ZoomName>

Defines the y-value of the centerpoint of the zoom area.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Position> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:ABSolute:SPAN <DiagramName>, <ZoomName>,<Span>
LAYout:ZOOM:VERTical:ABSolute:SPAN? <DiagramName>, <ZoomName>

Defines the height of the zoom area.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Span> Range: 0 to 100E+24
Increment: 0.01
*RST: 0.01

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:ABSolute:STARt <DiagGroupNme>,
<ZoomDiagramName>,<Start>
LAYout:ZOOM:VERTical:ABSolute:STARt? <DiagGroupNme>,
<ZoomDiagramName>

Defines the lower limit of the zoom area on the y-axis.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Start> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0

Parameters for setting and query:

<DiagGroupNme> Name of the diagram on which the zoom area is based.

<ZoomDiagramName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:ABSolute:STOP <DiagramName>, <ZoomName>,<Stop>
LAYout:ZOOM:VERTical:ABSolute:STOP? <DiagramName>, <ZoomName>

Defines the upper limit of the zoom area on the y-axis.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Stop> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:RELative:POSition <DiagramName>,
<ZoomName>,<RelPosi>
LAYout:ZOOM:VERTical:RELative:POSition? <DiagramName>, <ZoomName>

Defines the y-value of the centerpoint of the zoom area.

Parameters:

<RelPosi> Relative position of the centerpoint (y-value)

Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:RELative:SPAN <DiagramName>,
<ZoomName>,<RelativeSpan>
LAYout:ZOOM:VERTical:RELative:SPAN? <DiagramName>, <ZoomName>

Defines the height of the zoom area.

Parameters:
<RelativeSpan> Range: 1E-15 to 100
Increment: 0.1
*RST: 1
Default unit: %

Parameters for setting and query:
<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:RELative:STARt <DiagramName>,
<ZoomName>,<RelativeStart>
LAYout:ZOOM:VERTical:RELative:STARt? <DiagramName>, <ZoomName>

Defines the lower limit of the zoom area on the y-axis.

Parameters:
<RelativeStart> Range: 0 to 100
Increment: 0.1
*RST: 0
Default unit: %

Parameters for setting and query:
<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:VERTical:RELative:STOP <DiagramName>,
<ZoomName>,<RelativeStop>
LAYout:ZOOM:VERTical:RELative:STOP? <DiagramName>, <ZoomName>

Defines the upper limit of the zoom area on the y-axis.

Parameters:
<RelativeStop> Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Parameters for setting and query:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Usage: Asynchronous command

LAYout:ZOOM:REMove <DiagramName>, <ZoomName>

Removes the specified zoom diagram.

Setting parameters:

<DiagramName> Name of the diagram on which the zoom area is based.

<ZoomName> Name of the zoom diagram

Example: See Chapter 24.3.1.2, "Creating zoom diagrams", on page 1261

Usage: Setting only
Asynchronous command

24.10.2 Reference waveforms

- Reference....1446
- Scaling....1448
- Waveform data export....1453
- Import of multichannel waveform data....1454

24.10.2.1 Reference

REFCurve<m>:SOURce....1446
REFCurve<m>:STATe....1447
REFCurve<m>:NAME....1447
REFCurve<m>:OPEN....1447
REFCurve<m>:UPDate....1448
REFCurve<m>:SAVE....1448
REFCurve<m>:DELete....1448
REFCurve<m>:CLEar....1448

REFCurve<m>:SOURce <Source>

Selects the source waveform to be used as a reference.

Suffix:

<m> 1..4
Reference waveform

Parameters:

<Source> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | R1 | R2 | R3 | R4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Source of the reference waveform, see Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C1W1

Usage: Asynchronous command

REFCurve<m>:STATe <State>

If enabled, the reference waveform is displayed in the diagram.

Suffix:

<m> 1..4
Reference waveform

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

REFCurve<m>:NAME <Name>

Defines the name of the reference waveform file to be loaded, saved or deleted.

Suffix:

<m> 1..4
Reference waveform

Parameters:

<Name> Path and name of the file that contains the reference waveform or to which the reference waveform is to be stored (`.xml` or `.bin` format), enclosed in single quotes.

Usage: Asynchronous command

REFCurve<m>:OPEN

Loads the reference waveform file selected by `REFCurve<m>:NAME` on page 1447.

Suffix:

<m> 1..4
Reference waveform

Usage: Event

REFCurve<m>:UPDate

Copies the selected source waveform (see `REFCurve<m>:SOURce` on page 1446) with all its settings to the memory of the reference waveform.

Suffix:
<m> 1..4
Reference waveform

Usage: Event

REFCurve<m>:SAVE

Saves the reference waveform to the file selected by `REFCurve<m>:NAME` on page 1447.

Suffix:
<m> 1..4
Reference waveform

Usage: Event

REFCurve<m>:DELete

Deletes the reference waveform file selected by `REFCurve<m>:NAME` on page 1447.

Suffix:
<m> 1..4
Reference waveform

Usage: Event

REFCurve<m>:CLEar

The selected reference waveform is no longer displayed, its memory is deleted.

Suffix:
<m> 1..4
Reference waveform

Usage: Event
Asynchronous command

24.10.2.2 Scaling

REFCurve<m>:RESTore....1449
REFCurve<m>:VMODe....1449
REFCurve<m>:SCALe....1449
REFCurve<m>:POSition....1450
REFCurve<m>:TOORignal....1450
REFCurve<m>:RESCale:VERTical:STATe....1450
REFCurve<m>:RESCale:VERTical:FACTor....1450

REFCurve<m>:RESCale:VERTical:OFFSet....1451
REFCurve<m>:HMODe....1451
REFCurve<m>:RESCale:HORizontal:STATe....1452
REFCurve<m>:RESCale:HORizontal:FACTor....1452
REFCurve<m>:RESCale:HORizontal:OFFSet....1452
REFCurve<m>:RESCale:HORizontal:POSition....1453

REFCurve<m>:RESTore

Resets the time scale and the reference point to the original values of the reference waveform.

Suffix:
<m> 1..4
Reference waveform

Usage: Event
Asynchronous command

REFCurve<m>:VMODe <VerticalMode>

Selects the coupling of vertical settings.

Suffix:
<m> 1..4
Reference waveform

Parameters:
<VerticalMode> COUPled | INDependent

COUPled
Vertical position and scale of the source are used.

INDependent
Scaling and position can be set specific to the reference waveform.

*RST: INDependent

Usage: Asynchronous command

REFCurve<m>:SCALe <VerticalScale>

Sets the scale factor for the reference waveform if vertical scaling is set to "Independent" (see `REFCurve<m>:VMODe` on page 1449).

Suffix:
<m> 1..4
Reference waveform

Parameters:
<VerticalScale> Range: 1E-15 to 100E+24
Increment: 10E-6
*RST: 0.5
Default unit: V/div

Usage: Asynchronous command

REFCurve<m>:POSition <VertPosi>

Moves the reference waveform and its horizontal axis up or down in the diagram, if vertical scaling is set to "Independent" (see `REFCurve<m>:VMODe` on page 1449).

Suffix:
<m> 1..4
Reference waveform

Parameters:
<VertPosi> Range: -100E+24 to 100E+24
Increment: 0.02
*RST: 0
Default unit: div

Usage: Asynchronous command

REFCurve<m>:TOORignal

Restores the original vertical settings of the reference waveform (vertical scale, position, and offset), if vertical scaling is set to independent (see `REFCurve<m>:VMODe` on page 1449).

Suffix:
<m> 1..4

Usage: Event
Asynchronous command

REFCurve<m>:RESCale:VERTical:STATe <State>

Enables and disables the vertical stretching. Stretching changes the display of the waveform independent of the vertical scale and position.

Suffix:
<m> 1..4
Reference waveform

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

REFCurve<m>:RESCale:VERTical:FACTor <ScaleFactor>

Defines the vertical stretching factor. A factor greater than 1 stretches the waveform vertically, a factor lower than 1 compresses the curve.

Suffix:
<m> 1..4
Reference waveform

Parameters:
<ScaleFactor> Range: -1E+6 to 1E+6
Increment: 0.1
*RST: 1

Usage: Asynchronous command

REFCurve<m>:RESCale:VERTical:OFFSet <Offset>

Moves the reference waveform vertically. Like vertical offset of channel waveforms, the offset of a reference waveform is subtracted from the measured value.

Suffix:
<m> 1..4
Reference waveform

Parameters:
<Offset> Negative values shift the waveform up, positive values shift it down.
Range: -100E+24 to 100E+24
Increment: 1E-6
*RST: 0
Default unit: V

Usage: Asynchronous command

REFCurve<m>:HMODe <HorizontalMode>

Selects the coupling of horizontal settings.

Suffix:
<m> 1..4
Reference waveform

Parameters:
<HorizontalMode> ORIGinal | COUPled

ORIGinal
Horizontal scaling and reference point of the source waveform are used.

COUPled
The current horizontal settings of the diagram are used.

*RST: ORIGinal

Usage: Asynchronous command

REFCurve<m>:RESCale:HORizontal:STATe <State>

Enables and disables the horizontal stretching.

Stretching changes the display of the waveform independent of the horizontal settings of the source waveform and of the horizontal diagram settings.

Suffix:
<m> 1..4
Reference waveform

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

REFCurve<m>:RESCale:HORizontal:FACTor <ScaleFactor>

A factor greater than 1 stretches the waveform horizontally, a factor lower than 1 compresses the curve.

Suffix:
<m> 1..4
Reference waveform

Parameters:
<ScaleFactor> Range: 1E-6 to 1E+6
Increment: 0.1
*RST: 1

Usage: Asynchronous command

REFCurve<m>:RESCale:HORizontal:OFFSet <Offset>

Moves the waveform horizontally. Positive values shift the waveform to the right, negative values shift it to the left.

Suffix:
<m> 1..4
Reference waveform

Parameters:
<Offset> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0
Default unit: s

Usage: Asynchronous command

REFCurve<m>:RESCale:HORizontal:POSition <Offset>

Moves the waveform horizontally. Enter a value with a time unit suitable for the time scale of the diagram. Positive values shift the waveform to the right, negative values shift it to the left.

Suffix:
<m> 1..4

Parameters:
<Offset> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0
Default unit: s

Usage: Asynchronous command

24.10.2.3 Waveform data export

Commands for saving waveform data to file are described in Chapter 24.16.4, "Waveform data export to file", on page 1649.

Commands for reference waveforms are listed below.

REFCurve<m>:DATA:STYPe?.....1453
REFCurve<m>:DATA:HEADer?.....1453
REFCurve<m>:DATA[:VALues]?.....1454

REFCurve<m>:DATA:STYPe?

Returns the signal type of the source of the reference waveform.

Suffix:
<m> 1..4
Reference waveform

Return values:
<SignalType> SOUR | SPEC | CORR | NONE

SOURce = normal signal
SPECtrum = FFT spectrum, specific math signal
CORRelation = correlated signal, specific math signal
NONE = undefined

Usage: Query only

REFCurve<m>:DATA:HEADer?

Returns information on the reference waveform.

Table 24-4: Header data

Position	Meaning	Example
1	XStart in s	-9.477E-008 = - 94,77 ns
2	XStop in s	9.477E-008 = 94,77 ns
3	Record length of the waveform in Samples	200000
4	Number of values per sample interval. For reference waveforms the number is always 1.	1

Suffix:
<m> 1..4
Reference waveform

Example: `REFC:DATA:HEAD?`
`-9.477E-008,9.477E-008,200000,1`

Usage: Query only

REFCurve<m>:DATA[:VALues]?

Returns the data of the channel waveform points for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

To set the export format, use `FORMat[:DATA]`.

You can retrieve only Y-values, or X- and Y-values. Use `EXPort:WAVeform:INCXvalues` to define this.

Suffix:
<m> 1..4
Reference waveform

Return values:
<Data> List of values according to the format and content settings.

Usage: Query only

24.10.2.4 Import of multichannel waveform data

Commands for saving waveform data to file are described in Chapter 24.16.4, "Waveform data export to file", on page 1649.

Commands for reference waveforms are listed below.

REFCurve<m>:MULTichannel:NAME 1455
REFCurve<m>:MULTichannel:IMPort 1455
REFCurve<m>:MULTichannel:OPEN 1455

REFCurve<m>:MULTichannel:NAME <MltChImptPath>

Defines the path and the file to be imported. If not path is given, the default path `C:\Users\Public\Public Documents\Rohde-Schwarz\RTx\RefWaveforms` is used.

Suffix:
<m> 1..4
The suffix is irrelevant.

Parameters:
<MltChImptPath> String with path and filename

Example: See `REFCurve<m>:MULTichannel:IMPort` on page 1455.

Usage: Asynchronous command

REFCurve<m>:MULTichannel:IMPort <MltChImptWfmSel>

Assigns a waveform from the multichannel file to a reference waveform. To import all waveforms that are in the file, you must assign each waveform to another reference waveform.

Suffix:
<m> 1..4
Selects the reference waveform.

Parameters:
<MltChImptWfmSel> NONE | NONE | WF1 | WAVeform1 | WF2 | WAVeform2 | WF3 | WAVeform3 | WF4 | WAVeform4

WF1 = WAVeform1, WF2 = WAVeform2, WF3 = WAVeform3, WF4 = WAVeform4
Selects the waveform in the export file.

*RST: NONE

Example: Prerequisite: file `RefCurve_2016-02-16_01.bin` is a multi-channel export with at least WF1, WF3 and WF4.

```
REFCurve:MULTichannel:NAME 'RefCurve_2016-02-16_01.bin'
REFCurve1:MULTichannel:IMPort WF1
REFCurve2:MULTichannel:IMPort WF3
REFCurve3:MULTichannel:IMPort WF4
REFCurve:MULTichannel:OPEN
```

Usage: Asynchronous command

Firmware/software: FW 3.20

REFCurve<m>:MULTichannel:OPEN

Loads the waveform data to the reference waveforms.

Suffix:
<m> 1..4
The suffix is irrelevant.

Example: See `REFCurve<m>:MULTichannel:IMPort` on page 1455.

Usage: Event
Asynchronous command

24.10.3 Mathematics

CALCulate:MATH<m>[:EXPRession][:DEFine]....1456
CALCulate:MATH<m>:STATe....1456
CALCulate:MATH<m>:ENVSelection....1457
CALCulate:MATH<m>:ARIThmetics....1457
CALCulate:MATH<m>:VERTical:OFFSet....1458
CALCulate:MATH<m>:VERTical:RANGe....1458
CALCulate:MATH<m>:VERTical:SCALe....1459
CALCulate:MATH<m>:VERTical:MAXimum....1459
CALCulate:MATH<m>:DATA:STYPe?....1459
CALCulate:MATH<m>:DATA:HEADer?....1460
CALCulate:MATH<m>:DATA[:VALues]?....1460

CALCulate:MATH<m>[:EXPRession][:DEFine] <RemComplExpr>

Defines the math expression to be calculated for the specified math channel.

For an overview of corresponding expressions for the available keys in the formula editor, see Chapter 7.3.4, "Equation", on page 261.

Suffix:
<m> 1..8
Selects the math waveform.

Parameters:
<RemComplExpr> String with regular expression for calculation

Example: `CALC:MATH 'Ch1Wfm1*Ch2Wfm1'`
Defines the multiplication of waveforms Ch1Wfm1 and Ch2Wfm1.

Usage: Asynchronous command

CALCulate:MATH<m>:STATe <State>

Enables the math waveform display.

Suffix:
<m> 1..8
math waveform

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:ENVSelection <EnvelopeCurve>

Selects the upper or lower part of the waveform for mathematic calculation, or a combination of both.

The setting is relevant for waveforms with waveform arithmetic mode "Envelope" or with "Peak detect" decimation. All mathematic operations - except for derivation - can be applied to envelope waveforms and waveforms with "Peak detect" decimation.

Suffix:

<m> 1..8
Selects the math waveform.

Parameters:

<EnvelopeCurve> MIN | MAX | BOTH

*RST: BOTH

Usage: Asynchronous command

Firmware/software: FW 2.25

CALCulate:MATH<m>:ARIThmetics <Arithmetics>

Selects the method to build the resulting math waveform from consecutive acquisitions. The processing is similar to the waveform arithmetics - instead of the acquired waveforms the results of the mathematic formula are used to create envelope, average and RMS.

To define the reset condition for arithmetics calculation, use `ACQuire:ARESet:MODE`.

Suffix:

<m> 1..8
Selects the math waveform.

Parameters:

<Arithmetics> OFF | ENVelope | AVERage | RMS | MINHold | MAXHold

Waveform arithmetic mode

OFF
The math waveform is built according to the mathematic formula.

ENVelope
Detects the minimum and maximum math values in a sample interval over a number of acquisitions.

AVERage
Calculates the average from the math data of the current acquisition and a number of acquisitions before. To define the number of acquisitions, use `ACQuire:COUNt`.

RMS
The resulting math waveform is the root mean square of the current acquisition and a number of acquisitions before. The result is the average power spectrum. Number of acquisitions: `ACQuire:COUNt`

MAXHold
Determines the maximum result for each input value from the math data of the current acquisition and a number of acquisitions before. To define the number of acquisitions, use `ACQuire:COUNt`.

MINHold
Determines the minimum result for each input value from the math data of the current acquisition and a number of acquisitions before. To define the number of acquisitions, use `ACQuire:COUNt`.

*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:VERTical:OFFSet <VerticalOffset>

Sets a voltage offset to adjust the vertical position of the math function on the screen.

Suffix:
<m> 1..8
Math waveform

Parameters:
<VerticalOffset> Negative values move the waveform au, positive values move it down.

Range: -100E+12 to 100E+12
Increment: 0.01
*RST: 0
Default unit: div

Usage: Asynchronous command

CALCulate:MATH<m>:VERTical:RANGe <VerticalRange>

Defines the range of FFT values to be displayed.

Suffix:
<m> 1..8
Math waveform

Parameters:

<VerticalRange> Range: 0 to 1E+15
Increment: 0.01
*RST: 0
Default unit: div

Usage: Asynchronous command

CALCulate:MATH<m>:VERTical:SCALe <VerticalScale>

Defines the scale of the y-axis in the math function diagram. The value is defined as "V per division", e.g. *50V/div*. In this case, the horizontal grid lines are displayed in intervals of 50 V.

Suffix:

<m> 1..8
Math waveform

Parameters:

<VerticalScale> Range: 1E-12 to 100E+12
Increment: 10E-6
*RST: 0.5
Default unit: V/div

Usage: Asynchronous command

CALCulate:MATH<m>:VERTical:MAXimum <VerticalMax>

Defines the maximum value on y-axis for spectrum displays.

Suffix:

<m> 1..8

Parameters:

<VerticalMax> Range: -1E+15 to 1E+15
Increment: 0.01
*RST: 0
Default unit: div

Usage: Asynchronous command

CALCulate:MATH<m>:DATA:STYPe?

Returns the signal type of the source of the math waveform.

Suffix:

<m> 1..8
Selects the math waveform.

Return values:

<SignalType> SOUR | SPEC | CORR | MEAS | NONE

SOURce = normal signal
SPECtrum = FFT spectrum, specific math signal

CORRelation = correlated signal, specific math signal
MEAsurement = result of a measurement
NONE = undefined

Usage: Query only
Asynchronous command

CALCulate:MATH<m>:DATA:HEADer?

Returns the header of math waveform data. The header contains attributes of the waveform.

Table 24-5: Header data

Position	Meaning	Example
1	XStart in s	-9.477E-008 = - 94,77 ns
2	XStop in s	9.477E-008 = 94,77 ns
3	Record length of the waveform in Samples	200000
4	Number of values per sample interval. For most waveforms the result is 1, for peak detect and envelope waveforms it is 2. If the number is 2, the number of returned values is twice the number of samples (record length).	1

Suffix:
<m> 1..8
Selects the math waveform.

Example: `CALC:MATH1:DATA:HEAD?`
`-9.477E-008,9.477E-008,200000,1`

Usage: Query only

CALCulate:MATH<m>:DATA[:VALues]?

Returns the data of the math waveform points for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

To set the export format, use `FORMat[:DATA]`.

You can retrieve only Y-values, or X- and Y-values. Use `EXPort:WAVeform:INCXvalues` to define this.

Suffix:
<m> 1..8
Selects the math waveform.

Return values:
<Data> List of values according to the format and content settings.

Usage: Query only

24.10.4 History

CHANnel<m>[:WAVeform<n>]:HISTory[:STATe]....1461
ACQuire:AVAilable?....1461
CHANnel<m>[:WAVeform<n>]:HISTory:CURRent....1461
CHANnel<m>[:WAVeform<n>]:HISTory:STARt....1462
CHANnel<m>[:WAVeform<n>]:HISTory:STOP....1462
CHANnel<m>[:WAVeform<n>]:HISTory:TPACq....1463
CHANnel<m>[:WAVeform<n>]:HISTory:PLAY....1463
CHANnel<m>[:WAVeform<n>]:HISTory:REPLay....1463
CHANnel<m>[:WAVeform<n>]:HISTory:TSDate?....1464
CHANnel<m>[:WAVeform<n>]:HISTory:TSABsolute?....1464
CHANnel<m>[:WAVeform<n>]:HISTory:TSRelative?....1464
CHANnel<m>[:WAVeform<n>]:HISTory:TSRReference?....1465

CHANnel<m>[:WAVeform<n>]:HISTory[:STATe] <State>

Enables or disables the history display.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Parameters:

<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

ACQuire:AVAilable?

Returns the number of acquisitions that is saved in the memory. This number of acquisitions is available for history viewing. It is also the number of acquisitions in a fast segmentation acquisition series.

Return values:

<AcquisitionCount> Range: 0 to 4294967295

Usage: Query only
Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:CURRent <CurrAcqIdx>

Accesses a particular acquisition in the memory to display it. The query returns the index of the segment that is shown.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Parameters:

<CurrAcqIdx> History index: the newest segment has the index "0", older segments have a negative index: -(n-1), ... -1 , 0
where n is the number of acquired segments.

Range: 0 to -(n-1)
Increment: 1

Example:

```
CHAN2:HIST:STAT ON
CHAN2:HIST:CURR -1
*OPC
```

Displays the acquisition before last from the history.

Usage: Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:STARt <StrtAcqIdx>

Sets the index of the oldest history acquisition for the history viewing.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Parameters:

<StrtAcqIdx> The start index is always negative.

Range: 0 to -(n-1)
Increment: 1

Usage: Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:STOP <StpAcqIdx>

Sets the index of the latest segment to be displayed in the history viewer.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Parameters:

<StpAcqIdx> Index of the stop acquisition. The newest acquisition always has the index "0".

Range: 0 to -(n-1)
Increment: 1

Usage: Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:TPACq <TimePerAcq>

Sets the display time for one acquisition. The shorter the time, the faster is the replay.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Parameters:

<TimePerAcq>
Range: 40E-6 to 10
Increment: 1
*RST: 0.05
Default unit: s

Usage: Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:PLAY

Starts and stops the replay of the history waveforms.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Usage: Event
Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:REPLay <AutoRepeat>

If ON, the replay of the history waveform sequence repeats automatically. Otherwise, the replay stops at the stop index set with `CHANnel<m>[:WAVeform<n>]:HISTory:STOP`.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Parameters:
<AutoRepeat> ON | OFF
*RST: OFF

Usage: Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:TSDate?

Returns the date of the current acquisition that is shown in the history viewer (`CHANnel<m>[:WAVeform<n>]:HISTory:CURRent`).

Suffix:
<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Return values:
<DateAbsString> String with date of the current acquisition (absolute time)

Usage: Query only
Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:TSABsolute?

Returns the abolsute daytime of the current acquisition that is shown in the history viewer (`CHANnel<m>[:WAVeform<n>]:HISTory:CURRent`).

Suffix:
<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Return values:
<TimeAbsString> String containing the time and unit

Usage: Query only
Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:TSRelative?

Returns the relative time of the current acquisition - the time difference to the newest acquisition (index = 0).

See also: (`CHANnel<m>[:WAVeform<n>]:HISTory:CURRent`).

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Return values:

<TimeRelativ> Range: -100E+24 to 100E+24
Default unit: s

Usage: Query only
Asynchronous command

CHANnel<m>[:WAVeform<n>]:HISTory:TSRReference?

Returns the relative time of the currently selected acquisition and the internal reference time (horizontal alignment) in history view with respect to the acquisition with index 0.

Suffix:

<m> 1..4
Selects the input channel.

<n> 1..3
Selects the waveform. If [:WAVeform<n>] is omitted, waveform 1 is adressed.

Return values:

<TimeRelIntRef> Range: -100E+24 to 100E+24
Increment: 1
*RST: 0
Default unit: s

Usage: Query only
Asynchronous command

Firmware/software: Version 2.70

24.10.5 XY-Diagram

WAVeform<m>:XYCurve:RATio 1465
WAVeform<m>:XYCurve:STATe 1466
WAVeform<m>:XYCurve:SWAP 1466
LAYout:SIGNal:AXIS 1466
WAVeform<m>:XYCurve:XSOurce 1467
WAVeform<m>:XYCurve:YSOurce 1467

WAVeform<m>:XYCurve:RATio <ConstantXYRatio>

If enabled, the x- and y-axes maintain a constant ratio in the diagram.

Suffix:
<m> 1..4
XY-diagram

Parameters:
<ConstantXYRatio> ON | OFF
*RST: ON

Usage: Asynchronous command

WAVeform<m>:XYCurve:STATe <State>

Activates an XY-waveform.

Suffix:
<m> 1..4
XY-diagram

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

WAVeform<m>:XYCurve:SWAP

Replaces the source of the x-axis with the source of the y-axis and vice versa.

Suffix:
<m> 1..4
XY-diagram

Usage: Event
Asynchronous command

LAYout:SIGNal:AXIS <DiagramName>, <Source>, <XSource>

Creates an XY-diagram by adding a second waveform to a diagram.

Setting parameters:
<DiagramName> String with the name of the diagram where the waveform is added.

<Source> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL1 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Waveform to be added, see Chapter 24.4.2, "Waveform parameter", on page 1284

<XSource> ON | OFF

If on, the added waveform is assigned to the x-axis.
If off, it is assigned to the y-axis.

Usage: Setting only
Asynchronous command

WAVeform<m>:XYCurve:XSOurce <XSource>

Defines the signal source that supplies the x-values of the XY-diagram.

Suffix:
<m> 1..4
XY-diagram

Parameters:
<XSource> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG5TL1 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | SG4TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Source of x-values, see Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C1W1

Usage: Asynchronous command

WAVeform<m>:XYCurve:YSOurce <YSource>

Defines the signal source that supplies the y-values of the XY-diagram.

Suffix:
<m> 1..4
XY-diagram

Parameters:

<YSource> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Source of y-values, see Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C2W1

Usage: Asynchronous command

24.11 Cursor measurements

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

24.11.1 Cursor setup....1468
24.11.2 Cursor results....1476
24.11.3 Peak search using cursors....1477

24.11.1 Cursor setup

CURSor<m>:STATe....1469
CURSor<m>:AOFF....1469
CURSor<m>:FUNCtion....1469
CURSor<m>:SOURce....1470
CURSor<m>:SSOurce....1470
CURSor<m>:USSource....1471
CURSor<m>:TRACking[:STATe]....1472
CURSor<m>:X1Position....1472
CURSor<m>:X2Position....1472
CURSor<m>:XCOupling....1473
CURSor<m>:Y1Position....1473

CURSor<m>:Y2Position 1473
CURSor<m>:YCOupling 1474
CURSor<m>:X1ENvelope 1474
CURSor<m>:X2ENvelope 1475
CURSor<m>:LABel[:LABel] 1475
CURSor<m>:LABel:X<1..2> 1475
CURSor<m>:LABel:Y<1..2> 1476
CURSor<m>:SIAD 1476

CURSor<m>:STATe <State>

Switches the indicated cursor on or off.

Suffix:
<m> 1..4
Selects the cursor set.

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

CURSor<m>:AOFF

This command switches all cursors off.

Suffix:
<m> The numeric suffix is irrelevant.

Usage: Event
Asynchronous command

CURSor<m>:FUNCtion <Type>

Defines the type of the indicated cursor set.

Suffix:
<m> 1..4
Selects the cursor set.

Parameters:
<Type> HORizontal | VERTical | PAIRed

HORizontal
A pair of horizontal cursor lines.

VERTical
A pair of vertical cursor lines.

PAIRed
Both vertical and horizontal cursor line pairs.

*RST: PAIRed

Usage: Asynchronous command

CURSor<m>:SOURce <Source>

Defines the source of the cursor measurement.

Suffix:

<m> 1..4
Selects the cursor set.

Parameters:

<Source> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | XY1 | XY2 | XY3 | XY4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | MSOB1 | MSOB2 | MSOB3 | MSOB4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | SG1 | SG2 | SG3 | SG4 | SG5 | SG6 | SG7 | SG8 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL1 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Source of the cursor measurement, see Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C1W1

Usage: Asynchronous command

CURSor<m>:SSOurce <Source2>

Selects the second cursor source.

Suffix:

<m> 1..4
Selects the cursor set.

Parameters:

<Source2> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | XY1 | XY2 | XY3 | XY4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | MSOB1 | MSOB2 | MSOB3 | MSOB4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | SG1 | SG2 | SG3 | SG4 | SG5 | SG6 | SG7 | SG8 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL1 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2 | EYE1 | EYE2 | EYE3 | EYE4 | TDRZ | TDRH | TDRR | BATHtub1 | BMEas1 | SRESponse1 | SRTRack1 | SDTRack1 | TJTRack1 | DJTRack1 | DDJTrack1 | PJTRack1 | RJOTRack1 | TJSPectrum1 | DDJSpectrum1 | PJOSPectrum1 | RJSPectrum1 | BATHtub2 | BMEas2 | SRESponse2 | SRTRack2 | SDTRack2 | TJTRack2 | DJTRack2 | DDJTrack2 | PJTRack2 | RJTORack2 | TJSPectrum2 | DDJSpectrum2 | PJSPectrum2 | RJOSPectrum2 | SEYE1 | SEYE2 | SDHist1 | TJHist1 | DJHist1 | DDJHist1 | PJHist1 | RJOHist1 | SRHist2 | SDHist2 | TJHist2 | DJHist2 | DDJHist2 | PJHist2 | RJOHist2 | NBAThtub1 | NBMeas1 | TNHist1 | DNHist1 | DDNHist1 | PNHist1 | RNOHist1 | TNTRack1 | DNTRack1 | DDNTrack1 | PNTRack1 | RNOTRack1 | TNSPectrum1 | DDNSpectrum1 | PNSPectrum1 | RNSPectrum1 | NBAThtub2 | NBMeas2 | TNHist2 | DNHist2 | DDNHist2 | PNHist2 | RNHist2 | TNTRack2 | DNTRack2 | DDNTrack2 | PNTRack2 | RNOTRack2 | TNSPectrum2 | DDNSpectrum2 | PNSPectrum2 | RNOSPectrum2

Second source of the cursor measurement, see Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C2W1

Usage: Asynchronous command

Firmware/software: FW 3.60

CURSor<m>:USSource <UseSource2>

Enables the second cursor source. To select the second source, use `CURSor<m>:SSOurce` on page 1470.

If enabled, the second corsor lines Cx.2 measure on the second source. Using a second source, you can measure differences between two channels with cursors.

Suffix:

<m> 1..4
Selects the cursor set.

Parameters:
<UseSource2> ON | OFF
*RST: OFF

Usage: Asynchronous command

Firmware/software: FW 3.60

CURSor<m>:TRACking[:STATe] <TrackCurve>

If set to ON, the horizontal cursor lines follow the waveform.

Suffix:
<m> 1..4
Selects the cursor set.

Parameters:
<TrackCurve> ON | OFF
*RST: OFF

Usage: Asynchronous command

CURSor<m>:X1Position <XPosition1>

Defines the position of the left vertical cursor line.

Suffix:
<m> 1..4
Selects the cursor set.

Parameters:
<XPosition1> Range: 0 to 500
Increment: 0.1
*RST: 0
Default unit: s

Usage: Asynchronous command

CURSor<m>:X2Position <XPosition2>

Defines the position of the right vertical cursor line.

Suffix:
<m> 1..4
Selects the cursor set.

Parameters:
<XPosition2> Range: 0 to 500
Increment: 0.1
*RST: 0
Default unit: s

Usage: Asynchronous command

CURSor<m>:XCOupling <Coupling>

Defines the positioning mode of the vertical cursor.

Suffix:
<m> 1..4
Selects the cursor set.

Parameters:
<Coupling> ON | OFF

ON
Moving one cursor line moves the other cursor line too. The cursor lines always remain a fixed distance.

OFF
Each cursor line is positioned independently.

*RST: OFF

Usage: Asynchronous command

CURSor<m>:Y1Position <YPosition1>

Defines the position of the lower horizontal cursor line.

If `CURSor<m>:TRACking[:STATe]` is enabled, the query returns the measurement result - the lower vertical value of the waveform.

Suffix:
<m> 1..4
Selects the cursor set.

Parameters:
<YPosition1> Range: -50 to 50
Increment: 0.01
*RST: 0
Default unit: The unit depends on the type of the waveform.

Usage: Asynchronous command

CURSor<m>:Y2Position <YPosition2>

Defines the position of the upper horizontal cursor line.

If `CURSor<m>:TRACking[:STATe]` is enabled, the query returns the measurement result - the upper vertical value of the waveform.

Suffix:
<m> 1..4
Selects the cursor set.

Parameters:

<YPosition2> Range: -50 to 50
Increment: 0.01
*RST: 0
Default unit: The unit depends on the type of the waveform.

Usage: Asynchronous command

CURSor<m>:YCOupling <Coupling>

Defines the positioning mode of the horizontal cursor. If the horizontal cursor lines track the waveform, the y-coupling is irrelevant (`CURSor<m>:MODE TRACk`).

Suffix:

<m> 1..4
Selects the cursor set.

Parameters:

<Coupling> ON | OFF

ON
Moving one cursor line moves the other cursor line too. The cursor lines always remain a fixed distance.

OFF
Each cursor line is positioned independently.

*RST: OFF

Usage: Asynchronous command

CURSor<m>:X1ENvelope <EnvelopeCurve1>

If the waveform arithmetics are set to envelope curve (see `CHANnel<m>[:WAVeform<n>]:ARIThmetics` on page 1334) and `CURSor<m>:TRACking[:STATe]` is set to "ON", this setting defines how the first horizontal cursor is positioned.

Suffix:

<m> 1..*
Selects the cursor set. 4 cursor sets are available.

Parameters:

<EnvelopeCurve1> MIN | MAX

MIN
The horizontal cursor is set to the crossing point of the vertical cursor with the minimum waveform envelope.

MAX
The horizontal cursor is set to the crossing point of the vertical cursor with the maximum waveform envelope.

*RST: MIN

Usage: Asynchronous command

CURSor<m>:X2ENvelope <EnvelopeCurve2>

If the waveform arithmetics are set to envelope curve (see `CHANnel<m>[:WAVeform<n>]:ARIThmetics` on page 1334) and `CURSor<m>:TRACking[:STATe]` is set to "ON", this setting defines how the second horizontal cursor is positioned.

Suffix:
<m> 1..*
Selects the cursor set. 4 cursor sets are available.

Parameters:
<EnvelopeCurve2> MIN | MAX

MIN
The horizontal cursor is set to the crossing point of the vertical cursor with the minimum waveform envelope.

MAX
The horizontal cursor is set to the crossing point of the vertical cursor with the maximum waveform envelope.

*RST: MAX

Usage: Asynchronous command

CURSor<m>:LABel[:LABel] <ShowLabel>

Shows the cursor labels in the diagram.

Suffix:
<m> 1..4
Selects the cursor set.

Parameters:
<ShowLabel> ON | OFF

*RST: ON

Usage: Asynchronous command

CURSor<m>:LABel:X<1..2> <Name>

Defines a label to be displayed with the vertical cursors. X1 is the left cursor line, and X2 the right one.By default, the cursors are labeled as Cu1.X1, Cu1.X2, Cu2.X1, Cu2.X2, Cu3.X1, Cu3.X2, Cu4.X1, Cu4.X2.

Suffix:
<m> 1..4
Selects the cursor set.

Parameters:
<Name> String with the cursor label

Usage: Asynchronous command

CURSor<m>:LABel:Y<1..2> <Name>

Defines a label to be displayed with the vertical cursors. Y1 is the lower cursor line, and Y2 the upper one.By default, the cursors are labeled as Cu1.Y1, Cu1.Y2, Cu2.Y1, Cu2.Y2, Cu3.Y1, Cu3.Y2, Cu4.Y1, Cu4.Y2.

Suffix:
<m> 1..4
Selects the cursor set.

Parameters:
<Name> String with the cursor label

Usage: Asynchronous command

CURSor<m>:SIAD <ShwInAllDiags>

Shows the enabled cursor measurements in all active diagrams of the same domain (time/ spectrum).

Suffix:
<m> 1..*
The suffix is irrelevant. All cursor measurements are affected.

Parameters:
<ShowInAllDiagr> ON | OFF
*RST: ON

Usage: Asynchronous command

24.11.2 Cursor results

CURSor<m>:XDELta[:VALue]?....1476
CURSor<m>:XDELta:INVerse?....1477
CURSor<m>:YDELta[:VALue]?....1477
CURSor<m>:YDELta:SLOPe....1477

CURSor<m>:XDELta[:VALue]?

Queries the delta value (distance) of two vertical cursor lines.

Suffix:
<m> 1..4
Selects the cursor set.

Return values:
<Delta> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only
Asynchronous command

CURSor<m>:XDELta:INVerse?

Queries the inverse value of the delta value (distance) of the two vertical cursor lines.

Suffix:
<m> 1..4
Selects the cursor set.

Return values:
<DeltaInverse> Range: -100E+24 to 100E+24
*RST: 0
Default unit: Hz

Usage: Query only
Asynchronous command

CURSor<m>:YDELta[:VALue]?

Queries the delta value (distance) of the two horizontal cursor lines.

Suffix:
<m> 1..4
Selects the cursor set.

Return values:
<Delta> Range: -100E+24 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

CURSor<m>:YDELta:SLOPe <DeltaSlope>

Returns the inverse value of the voltage difference - the reciprocal of the vertical distance of two horizontal cursor lines: *1/ΔV*.

Suffix:
<m> 1..4
Selects the cursor set.

Parameters:
<DeltaSlope> Range: -100E+24 to 100E+24
Increment: 0
*RST: 0

Usage: Asynchronous command

24.11.3 Peak search using cursors

CURSor<m>:FFT:SETCenter 1478
CURSor<m>:FFT:TOCenter 1478
CURSor<m>:MAXimum[:PEAK] 1478

CURSor<m>:MAXimum:LEFT 1478
CURSor<m>:MAXimum:RIGHt 1478
CURSor<m>:MAXimum:NEXT 1479

CURSor<m>:FFT:SETCenter

Sets the center frequency to the frequency value that is measured at cursor line c1.

Suffix:
<m> The suffix is irrelevant.

Usage: Event
Asynchronous command

CURSor<m>:FFT:TOCenter

Sets the vertical cursor line c1 to the center frequency.

Suffix:
<m> The suffix is irrelevant.

Usage: Event
Asynchronous command

CURSor<m>:MAXimum[:PEAK]

Sets both cursors to the absolute peak value.

Suffix:
<m> 1..4
Selects the cursor set.

Usage: Event

CURSor<m>:MAXimum:LEFT

Sets cursor 2 to the next maximum to the left of the current position.

Suffix:
<m> 1..4
Selects the cursor set.

Usage: Event

CURSor<m>:MAXimum:RIGHt

Sets cursor 2 to the next peak to the right (from the current position).

Suffix:
<m> 1..4
Selects the cursor set.

Usage: Event

CURSor<m>:MAXimum:NEXT

Sets cursor 2 to the next smaller peak (from the current position).

Suffix:

<m> 1..4
Selects the cursor set.

Usage: Event

24.12 Automatic measurements

This chapter contains all remote commands to set up automatic measurements and to analyze the measurement results.

Selection of the measurement group: MEASurement<m>

With R&S RTO6, you can configure up to eight simultaneous measurement groups. In manual operation, these eight measurement groups are represented by subtabs "MG1" to "MG8" in the "Measurements" dialog box. For remote operation, the measurement group is indicated by the suffix MEAS<m>, containing the number of the measurement group.

Remote control: measurement suffix <m>	Manual operation: "MG" subtab
1 to 8 correspond to	"MG1" to "MG8"
9, 10: do not use	Not available

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

- General settings.....1480
- Results.....1485
- Amplitude/time measurements.....1489
- Arithmetic for amplitude/time and spectrum measurements.....1499
- Eye diagram measurements.....1508
- Spectrum.....1511
- Optical measurements.....1518
- Histograms.....1521
- Display.....1531

- Statistics and long-term measurements....1533
- Protocol....1539
- Track and trend....1545
- Gating....1546
- Limit check and event actions....1549
- Reference levels....1552

24.12.1 General settings

MEASurement<m>[:ENABle]....1480
MEASurement<m>:SOURce....1480
MEASurement<m>:FSRC....1482
MEASurement<m>:SSRC....1482
MEASurement<m>:CATegory....1482
MEASurement<m>:MAIN....1483
MEASurement<m>:ADDitional....1484
MEASurement<m>:AON....1485
MEASurement<m>:AOFF....1485

MEASurement<m>[:ENABle] <State>

Switches the indicated measurement on or off.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<State> ON | OFF
*RST: OFF

Example: See "Simple frequency and amplitude measurement" on page 1262.

Usage: Asynchronous command

MEASurement<m>:SOURce <SignalSource>, [<SignalSource2>]

Defines the source of the measurement. Availability of sources depends on the selected category and installed options.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<SignalSource> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | SBUS1 | SBUS2 | SBUS3 | SBUS4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL1 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2 | | SRESponse1 | SRTRack1 | SDTRack1 | TJTRack1 | DJTRack1 | DDJTrack1 | PJTRack1 | RJTRack1 | SRESponse2 | SRTRack2 | SDTRack2 | TJTRack2 | DJTRack2 | DDJTrack2 | PJTRack2 | RJTRack2 | SEYE1 | SEYE2 | TNTRack1 | DNTRack1 | DDNTrack1 | PNTRack1 | RNOTrack1 | TNTRack2 | DNTRack2 | DDNTrack2 | PNTRack2 | RNOTrack2 | PJDTrack1 | PJDTrack2 | PNDTrack1 | PNDTrack2

First source of the measurement, see Chapter 24.4.2, "Waveform parameter", on page 1284

<SignalSource2> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2 | | SRESponse1 | SRTRack1 | SDTRack1 | TJTRack1 | DJTRack1 | DDJTrack1 | PJTRack1 | RJTRack1 | SRESponse2 | SRTRack2 | SDTRack2 | TJTRack2 | DJTRack2 | DDJTrack2 | PJTRack2 | RJTRack2 | SEYE1 | SEYE2 | TNTRack1 | DNTRack1 | DDNTrack1 | PNTRack1 | RNOTrack1 | TNTRack2 | DNTRack2 | DDNTrack2 | PNTRack2 | RNOTrack2 | PJDTrack1 | PJDTrack2 | PNDTrack1 | PNDTrack2

Second measurement source. The 2nd source is required for amplitude/time measurements that are performed on two waveforms (e.g. delay, phase).
Digital channels are only available if <SignalSource> is a digital channel.
Step response, advanced jitter track components and synthetic eye sources are available for amplitude/time measurements only.

*RST: C1W1,C2W1

Example: See "Simple frequency and amplitude measurement" on page 1262.

MEASurement<m>:FSRC <Source>

Defines the first measurement source.

The command is an alternative to `MEASurement<m>:SOURce`.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<Source> Source of the measurement, see `MEASurement<m>:SOURce` on page 1480.
*RST: C1W1

Usage: Asynchronous command

MEASurement<m>:SSRC <Source2>

Defines the second measurement source.

The command is an alternative to `MEASurement<m>:SOURce`.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<Source2> Second source of the measurement, see `MEASurement<m>:SOURce` on page 1480.
Digital channels are only available if a digital channel is set as first measurement source using `MEASurement<m>:FSRC`.
*RST: C2W1

Usage: Asynchronous command

MEASurement<m>:CATegory <Category>

Defines the measurement category.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<Category> AMPTime | JITTer | EYEJitter | SPECtrum | HISTogram | PROTocol | OPTical

AMPTime
Amplitude and time measurements

JITTer
Jitter measurements, available if one of the jitter options is installed.

EYEJitter
Eye diagram measurements

SPECtrum
Spectrum measurements

HISTogram
Histogram measurements

PROTocol
Protocol measurements

OPTical
Optical measurements

*RST: AMPTime

Example: See "Creating and reading histograms" on page 1263

Usage: Asynchronous command

MEASurement<m>:MAIN <MeasType>

Defines the measurement that is used as a source for math calculations and result analysis. This measurement is mandatory in the measurement group. Further measurements are added to the measurement group with `MEASurement<m>:ADDitional`.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<MeasType>

Amplitude/time measurements
HIGH | LOW | AMPLitude | MAXimum | MINimum | PDELta | MEAN | RMS | STDDev | POVershoot | NOVershoot | AREA | RTIMe | FTIMe | PPULse | NPULse | PERiod | FREQuency | PDCYcle | NDCYcle | CYCarea | CYCMean | CYCRms | CYCStddev | PULCnt | DELay | PHASe | BWIDth | PSWitching | NSWitching | PULSetrain | EDGecount | SHT | SHR | DTOTrigger | PROBemeter | SLERising | SLEFalling | ATAR
See Chapter 24.12.3, "Amplitude/time measurements", on page 1489.
*RST value for amplitude/time measurements: `AMPLitude`.

Jitter measurements
CCJitter | NCJitter | CCWidth | CCDutycycle | TIE | UINTerval | DRATe | SKWDelay | SKWPhase
Only available if option R&S RTO6-K12 is installed. See Chapter 24.22.1, "Jitter measurements (option R&S RTO6-K12)", on page 2479.

Eye diagram measurements
ERPercent | ERDB | EHEight | EWIDth | ETOP | EBASe | QFACtor | RMSNoise | SNRatio | DCDistortion | ERTime | EFTime | EBRate | EAMPlitude | PPJitter | STDJitter | RMSJitter
See Chapter 24.12.5, "Eye diagram measurements", on page 1508
*RST value for eye/jitter measurements: `ERPercent`.

Spectrum measurements
CPOWer | OBWidth | SBWidth | THD | THDPCT | THDA | THDU | THDR | PLISt | HAR | THDF | SPAR
See Chapter 24.12.6, "Spectrum", on page 1511.
*RST value for spectrum measurements: `CPOWer`.

Histogram measurements
WCOunt | WSAMples | HSAMples | HPEak | PEAK | UPEakvalue | LPEakvalue | HMAXimum | HMINimum | MEDian | MAXMin | HMEan | HSTDdev | M1STddev | M2STddev | M3STddev | MKPositive | MKNegative
See Chapter 24.12.8.2, "Histogram measurement", on page 1527.
*RST value for histogram measurements: `WCOunt`.

Protocol measurements
NONE | AUDio | F2F | F2A | FLDValue | GAP | MBITrate | SBITrate | BIDLe | FCNT | FEC | FER | CFER
Requires option R&S RTO6-K500
See Chapter 24.12.11, "Protocol", on page 1539

Optical measurements
OAP | OMA
See Chapter 24.12.7, "Optical measurements", on page 1518.

Example: See "Simple frequency and amplitude measurement" on page 1262.
See "Creating and reading histograms" on page 1263

Usage: Asynchronous command

MEASurement<m>:ADDitional <MeasType>, <State>
MEASurement<m>:ADDitional? <MeasType>

Adds additional measurements to the indicated measurement group. Only one measurement can be enabled or disabled per command. The query returns the state of the specified measurement.

Note that each measurement group must contain measurements from the same category. For example, if the main measurement of measurement group 1 is amplitude, then you cannot enable an eye width measurement in the same measurement group.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<State> ON | OFF

Enables or disables the measurement.

Parameters for setting and query:

<MeasType> See `MEASurement<m>:MAIN` on page 1483.

Usage: Asynchronous command

MEASurement<m>:AON

Enables all additional measurements in all categories of the indicated measurement.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Usage: Event
Asynchronous command

MEASurement<m>:AOFF

Disables all additional measurements in all categories of the indicated measurement.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Usage: Event
Asynchronous command

24.12.2 Results

MEASurement<m>:ARES? 1486
MEASurement<m>:ARNames 1486
MEASurement<m>:RESult[:ACTual]? 1487
MEASurement<m>:RESult:AVG? 1487
MEASurement<m>:RESult:EVTCount? 1487
MEASurement<m>:RESult:NPEak? 1487
MEASurement<m>:RESult:PPEak? 1487
MEASurement<m>:RESult:RELiability? 1487
MEASurement<m>:RESult:RMS? 1487
MEASurement<m>:RESult:WFMCount? 1487
MEASurement<m>:RESult:STDDev? 1487
MEASurement<m>:RESult:STARt? 1489
MEASurement<m>:RESult:STOP? 1489
MEASurement<m>:RESult:COUNt? 1489

MEASurement<m>:ARES?

Returns the results of all active measurements for the selected measurement group. If statistics are enabled, the instrument returns also statistical results.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Return values:
<sData> Result string

Example:
```
:MEASurement:ARES?
2.351778656126e-001,1.343873517787e-001,
-1.343873517787e-001
```

Example:
```
:MEASurement:ARNames 1
:MEASurement:ARES?
Amplitude: 2.351778656126e-001,Max:
1.343873517787e-001,Min: -1.343873517787e-001
```

Example:
```
:MEASurement:ARNames 1
:MEASurement:STATistics:ENABle 1
:MEASurement:ARES?
Amplitude:
2.312252964427e-001,2.608695652174e-001,
2.292490118577e-001,2.391822576775e-001,
2.392462568683e-001,5.533663458383e-003,12447,
12447,Max: 1.343873517787e-001,
1.343873517787e-001,1.324110671937e-001,
1.340297640714e-001, 1.340319233732e-001,
7.608360495310e-004,12447,12447, Min:
-1.324110671937e-001,-1.324110671937e-001,
-1.343873517787e-001,-1.340169449387e-001,
1.340191641476e-001,7.712812700636e-004,12447,
12447
Results: current, peak+, peak-, average, RMS,
standard deviation, event count, waveform count
```

Usage: Query only
Asynchronous command

MEASurement<m>:ARNames <Identifier>

Enables a prefix that indicates the measurement in the result string of the `MEASurement<m>:ARES?` command.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<Identifier> ON | OFF

*RST: OFF

Example:

```
:MEASurement:ARNames 1
:MEASurement:ARES?
Low: -1.185770750988e-001,Amplitude:
2.371541501976e-001,Max: 1.343873517787e-001
```

Usage: Asynchronous command

MEASurement<m>:RESult[:ACTual]? [<MeasType>]
MEASurement<m>:RESult:AVG? [<MeasType>]
MEASurement<m>:RESult:EVTCount? [<MeasType>]
MEASurement<m>:RESult:NPEak? [<MeasType>]
MEASurement<m>:RESult:PPEak? [<MeasType>]
MEASurement<m>:RESult:RELiability? [<MeasType>]
MEASurement<m>:RESult:RMS? [<MeasType>]
MEASurement<m>:RESult:WFMCount? [<MeasType>]
MEASurement<m>:RESult:STDDev? [<MeasType>]

Return the statistic results of the specified measurement. If no parameter is specified, the result of the main measurement is returned. The main measurement is defined using `MEASurement<m>:MAIN`.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RELiability: reliability of the measurement result
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

For a detailed description of the results, see "Selection of the measurement group: MEASurement<m>" on page 1479.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Query parameters:

<MeasType>

Amplitude/time measurements
HIGH | LOW | AMPLitude | MAXimum | MINimum | PDELta | MEAN | RMS | STDDev | POVershoot | NOVershoot | AREA | RTIMe | FTIMe | PPULse | NPULse | PERiod | FREQuency | PDCYcle | NDCYcle | CYCarea | CYCMean | CYCRms | CYCStddev | PULCnt | DELay | PHASe | BWIDth | PSWitching | NSWitching | PULSetrain | EDGecount | SETup | HOLD | SHR | DTOTrigger | PROBemeter | SLERising | SLEFalling | ATAR
See Chapter 24.12.3, "Amplitude/time measurements", on page 1489.

Jitter measurements (option R&S RTO6-K12)
CCJitter | NCJitter | CCWidth | CCDutycycle | TIE | UINTerval | DRATe | SKWDelay | SKWPhase
See Chapter 24.22.1, "Jitter measurements (option R&S RTO6-K12)", on page 2479

Eye diagram measurements
ERPercent | ERDB | EHEight | EWIDth | ETOP | EBASe | QFACtor | RMSNoise | SNRatio | DCDistortion | ERTime | EFTime | EBRate | EAMPlitude | PPJitter | STDJitter | RMSJitter | CPPercent
See Chapter 24.12.5, "Eye diagram measurements", on page 1508.

Spectrum measurements
CPOWer | OBWidth | SBWidth | THD | THDPCT | THDA | THDU | THDR | PLISt | HAR | THDF | SPAR
See Chapter 24.12.6, "Spectrum", on page 1511.

Histogram measurements
WCOunt | WSAMples | HSAMples | HPEak | PEAK | UPEakvalue | LPEakvalue | HMAXimum | HMINimum | MEDian | MAXMin | HMEan | HSTDdev | M1STddev | M2STddev | M3STddev | MKPositive | MKNegative
See Chapter 24.12.8.2, "Histogram measurement", on page 1527.

Protocol measurements
NONE | AUDio | F2F | T2F | FLDValue | GAP | MBITrate | SBITrate | BIDLe | FCNT | FEC | FER | CFER
Requires option R&S RTO6-K510 for type AUDio, or option R&S RTO6-K500.

Optical measurements
OAP | OMA
See Chapter 24.12.7, "Optical measurements", on page 1518.

Example: See "Simple frequency and amplitude measurement" on page 1262.
See "Creating and reading histograms" on page 1263

Usage: Query only

MEASurement<m>:RESult:STARt? [<MeasType>]
MEASurement<m>:RESult:STOP? [<MeasType>]

Return the start and stop times of the specified measurement. The parameter defines the measurement. If no parameter is specified, the result of the main measurement is returned. The main measurement is defined using `MEASurement<m>:MAIN`.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Query parameters:
<MeasType> See `MEASurement<m>:RESult:STDDev?` on page 1487.

Usage: Query only

MEASurement<m>:RESult:COUNt?

Returns the number of result groups that are returned by `MEASurement:RESult:ACTual? HAR`.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Query parameters:
<MeasType> HAR
The command is only relevant for harmonic search.

Return values:
Number Number of result groups

Example:
```
:MEASurement:RESult:ACTual? HAR
99.9;-6.000139236;199.8;-80.701713562;299.7;-15.528377533;
:MEASurement:RESult:COUNt? HAR
4
```

Usage: Query only

Firmware/software: Version 2.70

24.12.3 Amplitude/time measurements

The following table lists the <MeasType> parameter values with a short description.

In addition, the suffixes <n> for limit-check remote commands are listed.

For a detailed description, see Chapter 8.2.5.1, "Overview of amplitude/time measurements", on page 316.

Table 24-6: Amplitude and time measurements

<MeasType> parameter value (remote control)	**Measurement (manual operation)**	**Description, result**	**Suffix <n>**
HIGH	High	High signal level	1
LOW	Low	Low signal level	2
AMPLitude	Amplitude	Amplitude of the signal	3
MAXimum	Max	Maximum value of the waveform	4
MINimum	Min	Minimum value of the waveform	5
PDELta	Peak to peak	Peak-to-peak value of the waveform	6
MEAN	Mean	Mean value of the waveform	7
RMS	RMS	RMS (Root Mean Square) value of the voltage	8
STDDev	σ (S-dev)	Standard deviation of the waveform	9
POVershoot	Pos. overshoot	Positive overshoot of a square wave	10
NOVershoot	Neg. overshoot	Negative overshoot of a square wave	11
AREA	Area	Area beneath the waveform (integral)	12
RTIMe	Rise time	Rise time of the left-most rising edge of the waveform.	13
FTIMe	Fall time	Falling time of the left-most falling edge of the waveform.	14
PPULse	Pos. pulse	Width of a positive pulse – a rising edge followed by a falling edge. The measurement requires at least one complete period of a triggered signal.	15
NPULse	Neg. pulse	Width of a negative pulse – a falling edge followed by a rising edge. The measurement requires at least one complete period of a triggered signal.	16
PERiod	Period	Length of the left-most signal period of the waveform	17
FREQuency	Frequency	Frequency of the signal. The result is based on the period measurement.	18
PDCYcle	Pos. duty cycle	Positive duty cycle. The measurement requires at least one complete period of a triggered signal.	19
NDCYcle	Neg. duty cycle	Negative duty cycle. The measurement requires at least one complete period of a triggered signal.	20
CYCarea	Cycle area	Area (integral) beneath one cycle	21
CYCMean	Cycle mean	Mean value of one cycle	22
CYCRms	Cycle RMS	The RMS (Root Mean Square) value of one cycle	23
CYCStddev	Cycle σ (S-dev)	Standard deviation of one cycle	24

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description, result	Suffix <n>
PULCnt	Pulse count	Number of positive or negative pulses of the waveform, or both	25
DELay	Delay	Time difference between any edges of two measurement sources at any reference level. The measurement result is negative if the edge of the second source comes before the edge of the first source.	26
PHASe	Phase	Phase difference between two waveforms	27
BWIDth	Burst width	Duration of one burst, measured from the first edge to the last	28
PSWitching	Pos. switching	Settling time at rising edges	29
NSWitching	Neg. switching	Settling time at falling edges	30
PULSetrain	Pulse train	Duration of N positive pulses, measured from the rising edge of the first pulse to the falling edge of the N-th pulse. N has to be configured.	31
EDGecount	Edge count	Number of positive or negative edges of the waveform, or both	32
SETup	Setup time	Parameters to query the setup and hold times. Use these parameters only in following queries: • `MEASurement<m>:ARES?` • `MEASurement<n>:RESult:..` commands	33
HOLD	Hold time		
SHT	Setup/Hold time	Setting parameter to enable Setup/Hold time measurements. Use this parameter only as setting in: • `MEASurement<m>:MAIN` on page 1483 • `MEASurement<m>:ADDitional` on page 1484	35
SHR	Setup/Hold ratio	Setup/Hold ratio measurement. Setup/Hold ratio is the ratio of the setup time to the sum of hold and setup time: $T_{Setup} / (T_{Setup} + T_{Hold})$ Use this parameter as setting to activate the Setup/Hold ratio measurement in: • `MEASurement<m>:MAIN` on page 1483 • `MEASurement<m>:ADDitional` on page 1484 It is also used in the following queries: • `MEASurement<m>:ARES?` • `MEASurement<n>:RESult:..` Used also in queries with and commands.	36
Used for jitter measurements (limit checks) see Chapter 24.22.1, "Jitter measurements (option R&S RTO6-K12)", on page 2479.			37 to 45
DTOTrigger	Delay to trigger	Time between the trigger event and a following signal slope. High accuracy even if the trigger event is outside the acquisition data.	46
PROBemeter	Trig. ProbeMeter	DC voltage measured by the connected active R&S probe	47
SLERising	Slew rate on rising edge	Steepness of the rising edge: voltage difference between the lower and higher reference level, divided by the rise time.	48

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description, result	Suffix <n>
SLEFalling	Slew rate on falling edge	Steepness of the falling edge: voltage difference between the higher and lower reference level, divided by the fall time.	49
ATAR	Arithmetic for amplitude/time measurements	Basic arithmetic calculations with measurement values and scalars.	50

MEASurement<m>:ENVSelect 1492
MEASurement<m>:DETThreshold 1493
MEASurement<m>:AMPTime:ALEVel 1493
MEASurement<m>:AMPTime:PFSLope 1493
MEASurement<m>:AMPTime:PSLope 1494
MEASurement<m>:AMPTime:DELay<n>:DIRection 1494
MEASurement<m>:AMPTime:DELay<n>:ECOunt 1495
MEASurement<m>:AMPTime:DELay<n>:LSELect 1495
MEASurement<m>:AMPTime:DELay<n>:SLOPe 1495
MEASurement<m>:AMPTime:PTCount 1496
MEASurement<m>:AMPTime:ESLope 1496
MEASurement<m>:AMPTime:CSLope 1496
MEASurement<m>:AMPTime:CLCK<n>:LSELect 1497
MEASurement<m>:AMPTime:DATA<n>:LSELect 1497
MEASurement<m>:AMPTime:DTOTrigger<n>:SLOPe 1497
MEASurement<m>:AMPTime:DTOTrigger<n>:LSELect 1498
MEASurement<m>:AMPTime:LCHeck<n>:VALid 1498
MEASurement<m>:AMPTime:LCHeck<n>:LOWer:LIMit 1499
MEASurement<m>:AMPTime:LCHeck<n>:UPPer:LIMit 1499
MEASurement<m>:AMPTime:LCHeck<n>:LOWer:MARGin 1499
MEASurement<m>:AMPTime:LCHeck<n>:UPPer:MARGin 1499

MEASurement<m>:ENVSelect <EnvelopeCurve>

The command is only relevant for measurements on envelope waveforms. It selects the envelope to be used for measurement.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<EnvelopeCurve> MIN | MAX | BOTH

MIN: measures on the lower envelope
MAX: measures on the upper envelope
BOTH: the envelope is ignored and the waveform measured as usual

*RST: BOTH

Usage: Asynchronous command

MEASurement<m>:DETThreshold <SignDetectThres>

Defines the value above which measurement results are displayed. Values beneath the threshold are considered to be noise and they are ignored.

Suffix:
<m> 1..10
irrelevant

Parameters:
<SignDetectThres> Range: 0 to 50
Increment: 1
*RST: 5
Default unit: %

Usage: Asynchronous command

MEASurement<m>:AMPTime:ALEVel <AreaLevel>

Defines the reference level used to integrate the waveform.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<AreaLevel> Range: -100E+24 to 100E+24
Increment: 0
*RST: 0
Default unit: V

Usage: Asynchronous command

MEASurement<m>:AMPTime:PFSLope <PeriodSlope>

Selects the slope direction for frequency and period measurements.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<PeriodSlope> FIRSt | POSitive | NEGative | EITHer

POSitive | NEGative
Measures the time between rising or falling edges, respectively.

EITHer
In multiple measurements, the time is measured both between rising edges and between falling edges.
In single measurements. the first edge is taken for the measurement.

FIRSt
Time is measured either between rising edges or between falling edges. The first edge is taken for the measurement. In single measurements, it works the same way as "Either".
Only available for analog measurement sources.

*RST: FIRSt (analog source), POSitive (digital source)

Usage: Asynchronous command

MEASurement<m>:AMPTime:PSLope <PulsesSlope>

Sets the first slope of the pulses to be counted. The setting is only relevant for pulse count measurement (`MEASurement<m>:MAIN PULCnt` or `MEASurement<m>:ADDitional PULCnt,ON`).

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<PulsesSlope> POSitive | NEGative | EITHer

Count either positive or negative pulses, or both.

*RST: POSitive

Usage: Asynchronous command

MEASurement<m>:AMPTime:DELay<n>:DIRection <EdgeCntDirct>

Selects the direction for counting slopes for each source: from the beginning of the waveform, or from the end.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2
Selects the source number.

Parameters:
<EdgeCntDirct> FRFI | FRLA

FRFI - FRom FIrst, counting starts with the first edge of the waveform.
FRLA - FRom LAst, counting starts with the last edge of the waveform.

*RST: FRFI

Usage: Asynchronous command

MEASurement<m>:AMPTime:DELay<n>:ECOunt <EdgeIndex>

Sets the number of the edge that is relevant for delay measurement for each source.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2
Selects the source number.

Parameters:

<EdgeIndex> Edge number

Range: 1 to 100000
Increment: 1
*RST: 1

Usage: Asynchronous command

MEASurement<m>:AMPTime:DELay<n>:LSELect <DlyLevSel>

Selects the reference level on which the time is measured for each source.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2
Selects the source number.

Parameters:

<DlyLevSel> UPPer | MIDDle | LOWer

*RST: MIDDle

Usage: Asynchronous command

MEASurement<m>:AMPTime:DELay<n>:SLOPe <Slope>

Sets the edge of each source, between which the delay is measured.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2
Selects the source number.

Parameters:

<Slope> POSitive | NEGative | EITHer

*RST: POSitive

Usage: Asynchronous command

MEASurement<m>:AMPTime:PTCount <PulseCount>

Sets the number of positive pulses for the pulse train measurement. It measures the duration of N positive pulses from the rising edge of the first pulse to the falling edge of the N-th pulse.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<PulseCount> Range: 1 to 2147483647
Increment: 1
*RST: 1

Usage: Asynchronous command

MEASurement<m>:AMPTime:ESLope <EdgesSlope>

Sets the edge direction to be counted: rising edges, falling edges, or both. The setting is only relevant for edge count measurement (`MEASurement<m>:MAIN EDGecount` or `MEASurement<m>:ADDitional EDGecount,ON`).

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<EdgesSlope> POSitive | NEGative | EITHer
*RST: POSitive

Usage: Asynchronous command

MEASurement<m>:AMPTime:CSLope <SetHoldClkSlp>

Sets the edge of the clock from which the setup and hold times are measured.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<SetHoldClkSlp> POSitive | NEGative | EITHer

EITHer
The clock edges next to the data edge are considered regardless of the clock slope.

*RST: POSitive

Usage: Asynchronous command

MEASurement<m>:AMPTime:CLCK<n>:LSELect <ClkLevSel>

Selects the reference level of the clock on which the time is measured. Reference level and clock slope define the time point for setup and hold measurements.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2
The suffix is irrelevant, omit it.

Parameters:

<ClkLevSel> UPPer | MIDDle | LOWer

*RST: MIDDle

Usage: Asynchronous command

MEASurement<m>:AMPTime:DATA<n>:LSELect <DatLevSel>

Selects the reference level of the data on which the setup and hold time are measured.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2
The suffix is irrelevant, omit it.

Parameters:

<DatLevSel> UPPer | MIDDle | LOWer

*RST: MIDDle

Usage: Asynchronous command

MEASurement<m>:AMPTime:DTOTrigger<n>:SLOPe <DlyTrigSlp>

Sets the edge direction to be used for delay measurement.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2
The suffix is irrelevant.

Parameters:

<DlyTrigSlp> POSitive | NEGative | EITHer

*RST: POSitive

Usage: Asynchronous command

MEASurement<m>:AMPTime:DTOTrigger<n>:LSELect <DlyTrigLevSel>

Selects the reference level of the measurement source on which the delay is measured.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2
The suffix is irrelevant.

Parameters:

<DlyTrigLevSel> UPPer | MIDDle | LOWer

*RST: MIDDle

Usage: Asynchronous command

MEASurement<m>:AMPTime:LCHeck<n>:VALid <ValidRange>

Enables or disables limit checking for amplitude vs. time measurements in the specified measurement channel.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..50
Number of the amplitude/time measurement, see Table 24-6.

Parameters:

<ValidRange> ILIMit | ULIMit | LLIMit | OLIMit

ILIMit
Inside (within) limit; between the upper and lower limit values

ULIMit
Upper limit; above the upper limit value

LLIMit
Lower limit; below the lower limit value

OLIMit
Outside limit; above the upper limit or below the lower limit values

*RST: ILIMit

Usage: Asynchronous command

MEASurement<m>:AMPTime:LCHeck<n>:LOWer:LIMit <LowerLimit>
MEASurement<m>:AMPTime:LCHeck<n>:UPPer:LIMit <UpperLimit>

Define the lower and upper limit for limit checking, respectiveley. The valid range is defined using the `MEASurement<m>:AMPTime:LCHeck<n>:VALid` command.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..50
Number of the amplitude/time measurement, see Table 24-6.

Parameters:

<UpperLimit> Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

MEASurement<m>:AMPTime:LCHeck<n>:LOWer:MARGin <LowerMargin>
MEASurement<m>:AMPTime:LCHeck<n>:UPPer:MARGin <UpperMargin>

Define the lower and upper margins for the limit check, respectively. Margins are not as strict as limits and must be within the valid value range. The valid range is defined using the `MEASurement<m>:AMPTime:LCHeck<n>:VALid` command.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..50
Number of the amplitude/time measurement type, see Table 24-6.

Parameters:

<UpperMargin> Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

24.12.4 Arithmetic for amplitude/time and spectrum measurements

MEASurement<m>:AMPTime:ARIThmetic:CONStant 1500
MEASurement<m>:AMPTime:ARIThmetic:COUNit 1500
MEASurement<m>:AMPTime:ARIThmetic:CUNState 1501
MEASurement<m>:AMPTime:ARIThmetic:CUUNit 1501
MEASurement<m>:AMPTime:ARIThmetic:CUUState 1502
MEASurement<m>:AMPTime:ARIThmetic:OPERator 1502
MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:AMPTime 1502

MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:GROup....1503
MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:SPECtrum....1503
MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:TYPE....1504
MEASurement<m>:SPECtrum:ARIThmetic:CONStant....1504
MEASurement<m>:SPECtrum:ARIThmetic:COUNit....1505
MEASurement<m>:SPECtrum:ARIThmetic:CUNState....1505
MEASurement<m>:SPECtrum:ARIThmetic:CUUNit....1505
MEASurement<m>:SPECtrum:ARIThmetic:CUUState....1506
MEASurement<m>:SPECtrum:ARIThmetic:OPERator....1506
MEASurement<m>:SPECtrum:ARIThmetic:PARameter<n>:AMPTime....1506
MEASurement<m>:SPECtrum:ARIThmetic:PARameter<n>:GROup....1507
MEASurement<m>:SPECtrum:ARIThmetic:PARameter<n>:SPECtrum....1507
MEASurement<m>:SPECtrum:ARIThmetic:PARameter<n>:TYPE....1508

MEASurement<m>:AMPTime:ARIThmetic:CONStant <ArithConstant>

Sets a constant numeric value as the second operand of the amplitude/time arithmetic measurement.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<ArithConstant> Range: 0 to 0
Increment: 1
*RST: 0

Usage: Asynchronous command

MEASurement<m>:AMPTime:ARIThmetic:COUNit <ArithCstUnt>

Selects the operation type to be performed on the selected operands for the amplitude/time arithmetic measurement.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<ArithCstUnt> DBM | DB | DB_HZ | DB_GHZ | DBC | DBC_HZ | DBPW_HZ | DBW | DBPW | DBM_HZ | DBMW | W | MW | W_HZ | J | V_S | A_S | DBV | DBMV | DBMV_HZ | DBMV_MHZ | DBUV | DBUV_HZ | DBUV_SQRT_HZ | DBUV_MHZ | DBUV_M | DBMV_M_HZ | DBUV_M_MHZ | V | MV | UV | VPP | VV | UV_HZ | UV_M_HZ | V_SQRT_HZ | DBA | DBMA | DBMA_HZ | DBMA_MHZ | DBUA | DBUA_HZ | DBUA_SQRT_HZ | DBUA_MHZ | DBUA_M | DBUA_M_HZ | DBUA_M_MHZ | A | UA_HZ | UA_M_HZ | A_SQRT_HZ | PCT | DBPT | DBPT_HZ | H | PH_GHZ | F | FF_GHZ | OHM | SIEMENS | S | DBPS | DBNS | DBUS | DBMS | DBS | HZ | DBHZ | DBKHZ | DBMHZ | DBGHZ | M | IN | DEG | RAD | PCT | SYMB | VA_LIN | VA_LOG | NONE | IRE | MILES | SYMB_S | MSYMB_S | MBIT_S | BIT_S | BIT | WORD | PPM | DB | PER_SEC | S_DIV | HZ_DIV | V_DIV | A_DIV | OHM_DIV | NONE_DIV | DIV | BER | S | S_S | BAUD | TDEG | VS | USER | UINV | W_DIV | V_V | A_V | W_V | BYTS | V_A | V_W | VA | VAR | UI | LN

*RST: NONE

Usage: Asynchronous command

MEASurement<m>:AMPTime:ARIThmetic:CUNState <ArithCustomNmeSt>

Enables the use of user-defined custom name for the amplitude/time arithmetic measurement.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<ArithCustomNmeSt> ON | OFF

*RST: OFF

Usage: Asynchronous command

MEASurement<m>:AMPTime:ARIThmetic:CUUNit <ArithCustomUnit>

Sets a user-defined value of the unit for the amplitude/time arithmetic measurement.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<ArithCustomUnit>

Usage: Asynchronous command

MEASurement<m>:AMPTime:ARIThmetic:CUUState <ArithCustomUntSt>

Enables the use of user-defined custom unit for the amplitude/time arithmetic measurement. You can set the unit with `MEASurement<m>:AMPTime:ARIThmetic:CUUNit`.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<ArithCustomUntSt> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:AMPTime:ARIThmetic:OPERator <ArithOperator>

Selects the operation type to be performed on the selected operands for the amplitude/time arithmetic measurement.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<ArithOperator> ADD | SUB | MUL | DIV
*RST: ADD

Usage: Asynchronous command

MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:AMPTime <ArithOp1MeasAmplitudeTime>

Selects the amplitude/time arithmetics measurements for the measurment group.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2
Selects the operand.

Parameters:

<Op1MeasAmpTime> HIGH | LOW | AMPLitude | MAXimum | MINimum | PDELta | MEAN | RMS | STDDev | POVershoot | NOVershoot | AREA | RTIMe | FTIMe | PPULse | NPULse | PERiod | FREQuency | PDCYcle | NDCYcle | CYCarea | CYCMean | CYCRms | CYCStddev | PULCnt | DELay | PHASe | BWIDth | PSWitching | NSWitching | PULSetrain | EDGecount | SETup | HOLD | SHT | SHR | CCJitter | NCJitter | CCWidth | CCDutycycle | TIE | UINTerval | DRATe | SKWDelay | SKWPhase | DTOTrigger | PROBemeter | SLERising | SLEFalling | ATAR

See Table 24-6.

*RST: HIGH

Usage: Asynchronous command

MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:GROup
<ArithOp2MeasGroup>

Sets the measurement group for the second operand of the amplitude/time arithmetic measurement.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 2

Parameters:

<Op2MeasGroup> MEAS1 | MEAS2 | MEAS3 | MEAS4 | MEAS5 | MEAS6 | MEAS7 | MEAS8 | IMEAS | IMEas | QUICK | QUICk

*RST: MEAS1

Usage: Asynchronous command

MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:SPECtrum
<ArithOp1MeasSpectrum>

Selects the spectrum arithmetics measurements for the measurment group.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2
Selects the operand.

Parameters:

<Op1MeasSpectrum> CPOWer | ACPower | OBWidth | SBWidth | TOI | AMMod | THD | THDPCT | THDPct | THDA | THDU | THDR | PLISt | HAR | THDF | SPAR

*RST: CPOWer

Usage: Asynchronous command

MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:TYPE <ArithOp2Type>

Sets the type of amplitude arithmetic measurement.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 2

Parameters:

<ArithOp2Type> MEASurement | CONStant

MEASurement
Performs an arithmetic operation between two measurement results.

CONStant
Performs an operation between a measurement result and a user defined fixed constant.

*RST: MEASurement

Usage: Asynchronous command

MEASurement<m>:SPECtrum:ARIThmetic:CONStant <ArithConstant>

Sets a constant numeric value as the second operand of the spectrum arithmetic measurement.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<ArithConstant> Range: - 3.4 e38 to + 3.4 e38
Increment: 1
*RST: 0

Usage: Asynchronous command

MEASurement<m>:SPECtrum:ARIThmetic:COUNit <ArithCstUnt>

Sets the unit for the constant, set with `MEASurement<m>:SPECtrum:ARIThmetic:CONStant`.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<ArithCstUnt> DBM | DB | DB_HZ | DB_GHZ | DBC | DBC_HZ | DBPW_HZ | DBW | DBPW | DBM_HZ | DBMW | W | MW | W_HZ | J | V_S | A_S | DBV | DBMV | DBMV_HZ | DBMV_MHZ | DBUV | DBUV_HZ | DBUV_SQRT_HZ | DBUV_MHZ | DBUV_M | DBMV_M_HZ | DBUV_M_MHZ | V | MV | UV | VPP | VV | UV_HZ | UV_M_HZ | V_SQRT_HZ | DBA | DBMA | DBMA_HZ | DBMA_MHZ | DBUA | DBUA_HZ | DBUA_SQRT_HZ | DBUA_MHZ | DBUA_M | DBUA_M_HZ | DBUA_M_MHZ | A | UA_HZ | UA_M_HZ | A_SQRT_HZ | PCT | DBPT | DBPT_HZ | H | PH_GHZ | F | FF_GHZ | OHM | SIEMENS | S | DBPS | DBNS | DBUS | DBMS | DBS | HZ | DBHZ | DBKHZ | DBMHZ | DBGHZ | M | IN | DEG | RAD | PCT | SYMB | VA_LIN | VA_LOG | NONE | IRE | MILES | SYMB_S | MSYMB_S | MBIT_S | BIT_S | BIT | WORD | PPM | DB | PER_SEC | S_DIV | HZ_DIV | V_DIV | A_DIV | OHM_DIV | NONE_DIV | DIV | BER | S | S_S | BAUD | TDEG | VS | USER | UINV | W_DIV | V_V | A_V | W_V | BYTS | V_A | V_W | VA | VAR | UI | LN

*RST: NONE

Usage: Asynchronous command

MEASurement<m>:SPECtrum:ARIThmetic:CUNState <ArithCustomNmeSt>

Enables the use of user-defined custom name for the spectrum arithmetic measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<ArithCustomNmeSt> ON | OFF

*RST: OFF

Usage: Asynchronous command

MEASurement<m>:SPECtrum:ARIThmetic:CUUNit <ArithCustomUnit>

Sets a user-defined value of the unit for the spectrum arithmetic measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<ArithCustomUnit>

Usage: Asynchronous command

MEASurement<m>:SPECtrum:ARIThmetic:CUUState <ArithCustomUntSt>

Enables the use of user-defined custom unit for the amplitude/time arithmetic measurement. You can set the unit with `MEASurement<m>:AMPTime:ARIThmetic:CUUNit`.

Suffix:
<m> 1..10

Parameters:
<ArithCustomUntSt> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:SPECtrum:ARIThmetic:OPERator <ArithOperator>

Selects the operation type to be performed on the selected operands for the spectrum arithmetic measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<ArithOperator> ADD | SUB | MUL | DIV
*RST: ADD

Usage: Asynchronous command

MEASurement<m>:SPECtrum:ARIThmetic:PARameter<n>:AMPTime <ArithOp1MeasAmplitudeTime>

Selects the amplitude/time arithmetics measurements for the measurment group.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2
Selects the operand.

Parameters:

<Op1MeasAmpTime> HIGH | LOW | AMPLitude | MAXimum | MINimum | PDELta | MEAN | RMS | STDDev | POVershoot | NOVershoot | AREA | RTIMe | FTIMe | PPULse | NPULse | PERiod | FREQuency | PDCYcle | NDCYcle | CYCarea | CYCMean | CYCRms | CYCStddev | PULCnt | DELay | PHASe | BWIDth | PSWitching | NSWitching | PULSetrain | EDGecount | SETup | HOLD | SHT | SHR | CCJitter | NCJitter | CCWidth | CCDutycycle | TIE | UINTerval | DRATe | SKWDelay | SKWPhase | DTOTrigger | PROBemeter | SLERising | SLEFalling | ATAR

*RST: HIGH

Usage: Asynchronous command

MEASurement<m>:SPECtrum:ARIThmetic:PARameter<n>:GROup
<ArithOp2MeasGroup>

Sets the measurement group for the second operand of the spectrum arithmetic measurement.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 2

Parameters:

<ArithOp2MeasGroup> MEAS1 | MEAS2 | MEAS3 | MEAS4 | MEAS5 | MEAS6 | MEAS7 | MEAS8 | IMEAS | IMEas | QUICK | QUICk

*RST: MEAS1

Usage: Asynchronous command

MEASurement<m>:SPECtrum:ARIThmetic:PARameter<n>:SPECtrum
<ArithOp1MeasSpectrum>

Selects the first operand for the sepctrum arithmetics measurements.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2
Selects the operand.

Parameters:

<Op1MeasSpectrum> CPOWer | ACPower | OBWidth | SBWidth | TOI | AMMod | THD | THDPCT | THDPct | THDA | THDU | THDR | PLISt | HAR | THDF | SPAR

See Table 24-8.

*RST: CPOWer

Usage: Asynchronous command

MEASurement<m>:SPECtrum:ARIThmetic:PARameter<n>:TYPE <ArithOp2Type>

Sets the type of spectrum arithmetic measurement.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 2

Parameters:

<ArithOp2Type> MEASurement | CONStant

MEASurement
Performs an arithmetic operation between two measurement results.

CONStant
Performs an operation between a measurement result and a user defined fixed constant.

*RST: MEASurement

Usage: Asynchronous command

24.12.5 Eye diagram measurements

The following table lists the <MeasType> parameter values with a short description.

In addition, the suffixes <n> for limit-check remote commands are listed.

For a detailed description, see Chapter 8.2.6.1, "Overview of eye diagram measurements", on page 332.

Table 24-7: Eye diagram measurements

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result	Suffix <n>
		not used	1
ERPercent	Extinction ratio (%)	Eye base / Eye top *100 Prerequisite: Eye base > 0 and Eye top > 0	2
ERDB	Extinction ratio (dB)	10*log (Eye top / Eye base)	3

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result	Suffix <n>
EHEight	Eye height	Vertical eye opening	4
EWIDth	Eye width	Horizontal eye opening	5
ETOP	Eye top	Mean of the upper vertical histogram	6
EBASe	Eye base	Mean of the lower vertical histogram	7
		not used	8
QFACtor	Q factor	(Eye top – Eye base) / (σtop + σbase)	10
		not used	11...13
RMSNoise	Noise (RMS)	Quadratic mean of the noise of eye top and eye base	14
SNRatio	S/N ratio	Signal-to-noise ratio 10 * log (Eye amplitude / Noise RMS)	15
DCDistortion	Duty cycle distortion	20 * log (Eye amplitude / Noise RMS)	16
ERTime	Eye rise time	Duration for signal to rise from 10% to 90% of the high signal level	17
EFTime	Eye fall time	Duration for signal to fall from 90% to 10% of the high signal level	18
EBRate	Eye bit rate	Frequency between two crossings	19
EAMPlitude	Eye amplitude	Eye top - Eye base	20
		not used	21...27
PPJitter	Jitter (peak to peak)	Average of the jitter for both crossing points (σcrossing1 + σcrossing2) / 2	28
STDJitter	Jitter (6*σ)	Jitter *6	29
RMSJitter	Jitter (RMS)	Quadratic mean of the jitter at both crossing points	30

MEASurement<m>:EYEJitter:AUToscale....1509
MEASurement<m>:EYEJitter:LCHeck<n>:VALid....1510
MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:LIMit....1510
MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:LIMit....1510
MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:MARGin....1511
MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:MARGin....1511

MEASurement<m>:EYEJitter:AUToscale

Defines optimized settings to perform an eye diagram measurement for the selected source.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Usage: Event
Asynchronous command

MEASurement<m>:EYEJitter:LCHeck<n>:VALid <ValidRange>

Enables or disables limit checking for eye/jitter measurements in the specified measurement channel.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..30
Number of the eye/jitter measurement, see Table 24-7.

Parameters:

<ValidRange> ILIMit | ULIMit | LLIMit | OLIMit

ILIMit
Inside (within) limit; between the upper and lower limit values

ULIMit
Upper limit; above the upper limit value

LLIMit
Lower limit; below the lower limit value

OLIMit
Outside limit; above the upper limit or below the lower limit values

*RST: ILIMit

Usage: Asynchronous command

MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:LIMit <LowerLimit>
MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:LIMit <UpperLimit>

Define the lower and upper limit for the limit check, respectively. The valid range is defined using the `MEASurement<m>:EYEJitter:LCHeck<n>:VALid` command.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..30
Number of the eye/jitter measurement, see Table 24-7.

Parameters:

<UpperLimit> Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:MARGin <LowerMargin>
MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:MARGin <UpperMargin>

Defines the upper margin for the limit check. Margins are not as strict as limits and must be within the valid value range. The valid range is defined using the `MEASurement<m>:EYEJitter:LCHeck<n>:VALid` command.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..30
Number of the eye/jitter measurement, see Table 24-7.

Parameters:

<UpperMargin> Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

24.12.6 Spectrum

The following table lists the <MeasType> parameter values with a short description.

In addition, the suffixes <n> for limit-check remote commands are listed.

For a detailed description, see Chapter 8.2.7.1, "Overview of spectrum measurements", on page 335.

Table 24-8: Spectrum measurements

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result	Suffix <n>
CPOWer	Channel power	Power integrated over the sample values defined by a center frequency and a bandwidth; based on a defined impedance; the result is given in mW	1
		not used	2
OBWidth	Occupied bandwidth	From the defined center frequency, symmetric sample value pairs to the left and right are integrated until a user-defined percentage of the total power is reached	3
SBWidth	Bandwidth	n dB down bandwidth. The samples to the left and right of the peak value are analyzed until the n dB threshold is exceeded. Frequencies at which the threshold is exceeded define the limits of the requested bandwidth.	4
		not used	5, 6
THD	THD[dB]	Total harmonic distortion in dB	7
THDPCT	THD[%]	Total harmonic distortion in %	8

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result	Suffix <n>
THDA	THD_a	Is equivalent to THD. Requires option R&S RTO6-K37.	9
THDU	THD_u	Requires option R&S RTO6-K37.	10
THDR	THD_r	Distortion factor. Requires option R&S RTO6-K37.	11
PLISt	Peak list	List of frequency and peak power value pairs. Requires option R&S RTO6-K37.	12
HAR	Harmonic search	Returns the measured harmonics. For each harmonic, the frequency and the value are listed. To get the number of result pairs (= harmonics), use `MEASurement<m>:RESult:COUNt?`.	13
THDF	THD_f	Root mean square of the sum of all amplitudes of the harmonic waves in relation to the amplitude at the fundamental frequency (first harmonic). Requires option R&S RTO6-K37.	14
SPAR	Arithmetic for spectrum measurements	Basic arithmetic calculations with measurement values and scalars.	15

MEASurement<m>:SPECtrum:CPOWer:BANDwidth 1512
MEASurement<m>:SPECtrum:OBANdwidth 1513
MEASurement<m>:SPECtrum:CPOWer:CFRequency 1513
MEASurement<m>:SPECtrum:NDBDown 1513
MEASurement<m>:SPECtrum:PEXCursion 1514
MEASurement<m>:SPECtrum:ATHReshold 1514
MEASurement<m>:SPECtrum:RESult<n>:COUNt 1514
MEASurement<m>:SPECtrum:RESult<n>:MODE 1515
MEASurement<m>:RESult:MAXCount 1515
MEASurement<m>:RESult:SHFRequency 1516
MEASurement<m>:RESult:SHLabels 1516
MEASurement<m>:SPECtrum:LCHeck<n>:VALid 1517
MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:LIMit 1517
MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:LIMit 1517
MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:MARGin 1518
MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:MARGin 1518

MEASurement<m>:SPECtrum:CPOWer:BANDwidth <ChPowBw>

Defines the bandwidth over which the channel power is calculated.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<ChPowBw>
Range: 0 to 4E+9
Increment: 1
*RST: 0
Default unit: Hz

Usage: Asynchronous command

MEASurement<m>:SPECtrum:OBANdwidth <OccupiedBW>

Defines the percentage of the total power used to determine the occupied bandwidth.

Suffix:

<m>
1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<OccupiedBW>
Range: 0.1 to 99.9
Increment: 1
*RST: 20
Default unit: %

Usage: Asynchronous command

MEASurement<m>:SPECtrum:CPOWer:CFRequency <ChPowCtrFreq>

Defines the center frequency from which the channel power is calculated over the specified bandwidth.

Suffix:

<m>
1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<ChPowCtrFreq>
Range: 0 to 6E+9
Increment: 1
*RST: 0
Default unit: Hz

Usage: Asynchronous command

MEASurement<m>:SPECtrum:NDBDown <NDbDown>

Defines the threshold until which the samples to the left and right of the peak value are analyzed in order to determine the "N dB down bandwidth".

Suffix:

<m>
1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<NDbDown> Range: 0 to 100
Increment: 1
*RST: 20
Default unit: dB

Usage: Asynchronous command

MEASurement<m>:SPECtrum:PEXCursion <Value>

Defines a relative threshold, the minimum level value by which the waveform must rise or fall to be considered as a peak. To avoid identifying noise peaks, enter a peak excursion value that is higher than the noise levels.

Suffix:

<m> 1..10
The suffix is irrelevant.

Parameters:

<Value>

Usage: Asynchronous command

Firmware/software: Version 2.70

MEASurement<m>:SPECtrum:ATHReshold <Value>

Defines an absolute threshold as an additional condition for the peak search. Only peaks that exceed the threshold are detected.

This setting is only available for spectrum waveforms. It is valid for cursor measurements, spectrum measurements and peak search.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<Value> numeric value
Default unit: dBm

Usage: Asynchronous command

MEASurement<m>:SPECtrum:RESult<n>:COUNt <MaxNoOfResults>

Sets the maximum number of measurement results that are listed in the result table. Available for peak list and harmonic search measurements (`MEASurement:MAIN` is set to `PLISt` or `HAR`).

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 12 | 13
12: Peak list measurements
13: Harmonic search measurements

Parameters:

<MaxNoOfResults> Range: 1 to 1000
Increment: 1
*RST: 10

Usage: Asynchronous command

MEASurement<m>:SPECtrum:RESult<n>:MODE <ResultMode>

Sets the way how the measurement results are displayed. Available only if `MEASurement:MAIN` is set to `PLISt` or `HAR`.

Suffix:

<m> 1..10

<n> 12 | 13
12: Peak list measurements
13: Harmonic search measurements

Parameters:

<ResultMode> ABS | REL

ABS
The harmonics/peaks are shown in absolute value, dBm.

REL
The level of the carrier is shown in absolute value dBm. The values the harmonics/peaks are shown relatively to the carrier in dBc.

*RST: ABS

Example: `:MEASurement1:SPECtrum:RESult13:MODE ABS`
The harmonic measurements are shown in absolute value, dBm.

Usage: Asynchronous command

MEASurement<m>:RESult:MAXCount <Number>
MEASurement<m>:RESult:MAXCount? <MeasType>

Defines the maximum number of peaks that are labeled in the diagram.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<Number> Maximum number of peaks

Query parameters:
<MeasType> PLISt

Only for peaklist measurements. For other measurement, this command returns an error.

MEASurement<m>:RESult:SHFRequency <MeasType>, <State>
MEASurement<m>:RESult:SHFRequency? <MeasType>

Includes the frequency of the detected peak in the diagram labels (if `MEASurement<m>:RESult:LABorder`=`ON`).

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<State> ON | OFF

Parameters for setting and query:
<MeasType> PLISt

Only for peaklist measurements. For other measurements, this command returns an error.

Example: Display labels.
```
MEAS:RES:SHL PLIS,ON
```
Show frequency in labels.
```
MEAS:RES:SHFR PLIS,ON
```
Query the type of labels for peak lists.
```
MEAS:RES:SHFR? PLIS
//Result: ON
```

MEASurement<m>:RESult:SHLabels <MeasType>, <State>
MEASurement<m>:RESult:SHLabels? <MeasType>

Defines whether a description (label) is displayed for each detected peak in the spectrum diagram.

The layout of the label is defined by `MEASurement<m>:RESult:LABorder`.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<State> ON | OFF

Parameters for setting and query:

<MeasType> PLISt

Only for peaklist measurements. For other measurements, this command returns an error.

Example: Display labels.

```
MEAS:RES:SHL PLIS,ON
```

MEASurement<m>:SPECtrum:LCHeck<n>:VALid <ValidRange>

Enables or disables limit checking for spectrum measurements in the specified measurement channel.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..15
Number of the spectrum measurement, see Table 24-8.

Parameters:

<ValidRange> ILIMit | ULIMit | LLIMit | OLIMit

ILIMit
Inside (within) limit; between the upper and lower limit values

ULIMit
Upper limit; above the upper limit value

LLIMit
Lower limit; below the lower limit value

OLIMit
Outside limit; above the upper limit or below the lower limit values

*RST: ILIMit

Usage: Asynchronous command

MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:LIMit <LowerLimit>
MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:LIMit <UpperLimit>

Define the lower and upper limits for the limit check, respectively. The valid range is defined using the `MEASurement<m>:SPECtrum:LCHeck<n>:VALid` command.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..15
Number of the spectrum measurement, see Table 24-8.

Parameters:

<UpperLimit> Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:MARGin <LowerMargin>
MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:MARGin <UpperMargin>

Define the lower and upper margins for the limit check, respectively. Margins are not as strict as limits and must be within the valid value range. The valid range is defined using the `MEASurement<m>:SPECtrum:LCHeck<n>:VALid` command.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..15
Number of the spectrummeasurement, see Table 24-8.

Parameters:

<UpperMargin> Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

24.12.7 Optical measurements

The following table lists the <MeasType> parameter values with a short description.

In addition, the suffixes <n> for limit-check remote commands are listed.

Table 24-9: Optical measurements

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result	Suffix <n>
OAP	Average power	True average component of an optical signal. $X_{Average} = \frac{1}{N_{Eval}} * \sum_{i=1}^{N_{Eval}} x(i)$	1
OMA	Modulation amplitude	Measure of the difference between the optical power of a non-return-to-zero (NRZ) one pulse and the optical power of an NRZ zero pulse. OMA requires an NRZ pattern.	2

MEASurement<m>:OPTical:OMAMplitude:RELative:STARt....1519
MEASurement<m>:OPTical:OMAMplitude:RELative:STOP....1519
MEASurement<m>:OPTical:UNIT....1519

MEASurement<m>:OPTical:LCHeck<n>:LOWer:LIMit 1520
MEASurement<m>:OPTical:LCHeck<n>:UPPer:LIMit 1520
MEASurement<m>:OPTical:LCHeck<n>:LOWer:MARGin 1520
MEASurement<m>:OPTical:LCHeck<n>:UPPer:MARGin 1520
MEASurement<m>:OPTical:LCHeck<n>:VALid 1520

MEASurement<m>:OPTical:OMAMplitude:RELative:STARt <Start>

Sets the start value of the range for the optical modulation amplitude.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<Start> Range: 1 to 99
Increment: 0.1
*RST: 40
Default unit: %

Usage: Asynchronous command

MEASurement<m>:OPTical:OMAMplitude:RELative:STOP <Stop>

Sets the end value of the range for the optical modulation amplitude.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<Stop> Range: 1 to 99
Increment: 0.1
*RST: 60
Default unit: %

Usage: Asynchronous command

MEASurement<m>:OPTical:UNIT <Unit>

Selects the unit for the optical measurements.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<Unit> WATT | DBM
*RST: WATT

Usage: Asynchronous command

MEASurement<m>:OPTical:LCHeck<n>:LOWer:LIMit <LowerLimit>
MEASurement<m>:OPTical:LCHeck<n>:UPPer:LIMit <UpperLimit>

Define the lower/ upper limit for limit checking of the optical measurements.

The range is defined with `MEASurement<m>:OPTical:LCHeck<n>:VALid`.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2

Parameters:
<UpperLimit> Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

MEASurement<m>:OPTical:LCHeck<n>:LOWer:MARGin <LowerMargin>
MEASurement<m>:OPTical:LCHeck<n>:UPPer:MARGin <UpperMargin>

Define the lower/ upper margin for limit checking of the optical measurements.

The range is defined with `MEASurement<m>:OPTical:LCHeck<n>:VALid`.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2

Parameters:
<UpperMargin> Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

MEASurement<m>:OPTical:LCHeck<n>:VALid <ValidRange>

Selects the condition for the limit check of the optical measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2

Parameters:

<ValidRange> ILIMit | ULIMit | LLIMit | OLIMit

`ILIMit`: Inside (within) limit; between the upper and lower limit values
`ULIMit`: Upper limit; above the upper limit value
`LLIMit` : Lower limit; below the lower limit value
`OLIMit`: Outside limit; above the upper limit or below the lower limit values

*RST: ILIMit

Usage: Asynchronous command

24.12.8 Histograms

See also: Chapter 24.16.5, "Waveform histogram export to file", on page 1656.

- Histogram display....1521
- Histogram measurement....1527

24.12.8.1 Histogram display

LAYout:HISTogram:ADD.... 1521
LAYout:HISTogram:ENABle....1522
LAYout:HISTogram:SOURce.... 1523
LAYout:HISTogram:MODE.... 1523
LAYout:HISTogram:HORZ:MODE.... 1524
LAYout:HISTogram:HORZ:ABSolute:STARt.... 1524
LAYout:HISTogram:HORZ:ABSolute:STOP....1524
LAYout:HISTogram:HORZ:RELative:STARt....1525
LAYout:HISTogram:HORZ:RELative:STOP.... 1525
LAYout:HISTogram:VERTical:MODE.... 1525
LAYout:HISTogram:VERTical:ABSolute:STARt.... 1526
LAYout:HISTogram:VERTical:ABSolute:STOP....1526
LAYout:HISTogram:VERTical:RELative:STARt....1526
LAYout:HISTogram:VERTical:RELative:STOP.... 1526
LAYout:HISTogram:RESet....1527
LAYout:HISTogram:REMove....1527

LAYout:HISTogram:ADD <HistogramName>, <Source>, <XStart>, <XStop>, <YStart>, <YStop>, <Relative>, <Orientation>

Defines and displays a new histogram for the specified source.

Note: To define the mode of the histogram (vertical or horizontal), use the `LAYout:HISTogram:MODE` command.

For numerical values, the input of a unit is not supported for this command.

Setting parameters:

<HistogramName> String defining the histogram name which is used to refer to the histogram by other functions.

<Source> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | MRESult1 | MRESult2 | MRESult3 | MRESult4 | MRESult5 | MRESult6 | MRESult7 | MRESult8 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2 | EYE1 | EYE2 | EYE3 | EYE4 | SRHist1 | SDHist1 | TJHist1 | DJHist1 | DDJHist1 | PJOHist1 | RJOHist1 | SEYE1 | SRHist2 | SDHist2 | TJHist2 | DJHist2 | DDJHist2 | PJHOist2 | RJOHist2 | SEYE2 | TNHist1 | DNHist1 | DDNHist1 | PNHist1 | RNOHist1 | TNHist2 | DNHist2 | DDNHist2 | PNHist2 | RNOHist2 | PJDHist1 | BUJHist1 | RJHist1 | PJDHist2 | BUJHist2 | RJHist2 | PNDHist1 | BUNHist1 | RNHist1 | PNDHist2 | BUNHist2 | RNHist2

Data source of the histogram, see Chapter 24.4.2, "Waveform parameter", on page 1284

<XStart> Defines the start value of the x-value range.

<XStop> Defines the stop value of the x-value range.

<YStart> Defines the start value of the y-value range.

<YStop> Defines the stop value of the y-value range.

<Relative> ON | OFF

Defines whether relative or absolute values are used for the value range definition.

<Orientation> VERTical | HORizontal

Example: See "Creating and reading histograms" on page 1263

Usage: Setting only
Asynchronous command

LAYout:HISTogram:ENABle <HistogramName>,<State>
LAYout:HISTogram:ENABle? <HistogramName>

Enables or disables the histogram evaluation and display.

Parameters:

<State> ON | OFF

*RST: ON

Parameters for setting and query:

<HistogramName> String parameter, name of the histogram as defined with `LAYout:HISTogram:ADD`.

Usage: Asynchronous command

LAYout:HISTogram:SOURce <HistogramName>,<Source>
LAYout:HISTogram:SOURce? <HistogramName>

Defines the waveform which is the source of the histogram.

Parameters:

<Source> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | MRESult1 | MRESult2 | MRESult3 | MRESult4 | MRESult5 | MRESult6 | MRESult7 | MRESult8 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2 | EYE1 | EYE2 | EYE3 | EYE4 | SRHist1 | SDHist1 | TJHist1 | DJHist1 | DDJHist1 | PJOHist1 | RJOHist1 | SEYE1 | SRHist2 | SDHist2 | TJHist2 | DJHist2 | DDJHist2 | PJOHist2 | RJOHist2 | SEYE2 | TNHist1 | DNHist1 | DDNHist1 | PNHist1 | RNOHist1 | TNHist2 | DNHist2 | DDNHist2 | PNHist2 | RNOHist2 | PJDHist1 | BUJHist1 | RJHist1 | RJHist2 | PNDHist1 | BUNHist1 | RNHist1 | PNDHist2 | BUNHist2 | RNHist2

Waveform source of the histogram, see Chapter 24.4.2, "Waveform parameter", on page 1284.
If the histogram source is a measurement, the histogram shows the density distribution of the results for the selected measurement (`MEAS<m>:MAIN`).

Parameters for setting and query:

<HistogramName> String parameter

Usage: Asynchronous command

LAYout:HISTogram:MODE <HistogramName>,<Mode>
LAYout:HISTogram:MODE? <HistogramName>

Defines or queries the type of histogram.

Parameters:

<Mode> VERTical | HORizontal

VERTical
Amplitude histogram (horizontal bars across amplitude)

HORizontal
Time or frequency histogram (vertical bars over time/frequencies)

*RST: VERTical

Parameters for setting and query:
<HistogramName> The name of the histogram as defined using `LAYout:HISTogram:ADD` on page 1521.

Usage: Asynchronous command

LAYout:HISTogram:HORZ:MODE <HistogramName>,<Mode>
LAYout:HISTogram:HORZ:MODE? <HistogramName>

Defines or queries whether the value range limits are entered as absolute or relative values.

Parameters:
<Mode> ABS | REL
*RST: ABS

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:HORZ:ABSolute:STARt <HistogramName>,<Start>
LAYout:HISTogram:HORZ:ABSolute:STARt? <HistogramName>

Defines the horizontal start value of the histogram.

For numerical values, the input of a unit is not supported for this command.

Parameters:
<Start> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:HORZ:ABSolute:STOP <HistogramName>,<Stop>
LAYout:HISTogram:HORZ:ABSolute:STOP? <HistogramName>

Defines the horizontal stop value of the histogram.

For numerical values, the input of a unit is not supported for this command.

Parameters:
<Stop> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:HORZ:RELative:STARt <HistogramName>,<RelativeStart>
LAYout:HISTogram:HORZ:RELative:STARt? <HistogramName>

Defines the horizontal start value of the histogram.

Parameters:
<RelativeStart> Range: 0 to 100
Increment: 0.1
*RST: 0
Default unit: %

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:HORZ:RELative:STOP <HistogramName>,<RelativeStop>
LAYout:HISTogram:HORZ:RELative:STOP? <HistogramName>

Defines the horizontal stop value of the histogram.

Parameters:
<RelativeStop> Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:VERTical:MODE <HistogramName>,<Mode>
LAYout:HISTogram:VERTical:MODE? <HistogramName>

Defines or queries whether the value range limits are entered as absolute or relative values.

Parameters:
<Mode> ABS | REL
*RST: ABS

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:VERTical:ABSolute:STARt <HistogramName>,<Start>
LAYout:HISTogram:VERTical:ABSolute:STARt? <HistogramName>

Defines the vertical start value of the histogram.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Start>	Range:	-100E+24 to 100E+24
	Increment:	0.01
	*RST:	0

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:VERTical:ABSolute:STOP <HistogramName>,<Stop>
LAYout:HISTogram:VERTical:ABSolute:STOP? <HistogramName>

Defines the vertical stop value of the histogram.

For numerical values, the input of a unit is not supported for this command.

Parameters:

<Stop>	Range:	-100E+24 to 100E+24
	Increment:	0.01
	*RST:	0.01

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:VERTical:RELative:STARt <HistogramName>,<RelativeStart>
LAYout:HISTogram:VERTical:RELative:STARt? <HistogramName>

Defines the vertical start value of the histogram.

Parameters:

<RelativeStart>	Range:	0 to 100
	Increment:	0.1
	*RST:	0
	Default unit:	%

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:VERTical:RELative:STOP <HistogramName>,<RelativeStop>
LAYout:HISTogram:VERTical:RELative:STOP? <HistogramName>

Defines the vertical stop value of the histogram.

Parameters:
<RelativeStop> Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Parameters for setting and query:
<HistogramName>

Usage: Asynchronous command

LAYout:HISTogram:RESet <HistogramName>

Resets the values to begin a new histogram.

Setting parameters:
<HistogramName>

Usage: Setting only
Asynchronous command

LAYout:HISTogram:REMove <Name>

Removes the specified histogram.

Setting parameters:
<Name>

Usage: Setting only
Asynchronous command

24.12.8.2 Histogram measurement

This chapter lists commands to set up measurements on histograms.

Note that the R&S RTO6 performs histogram measurements on the measurement that is defined with `MEASurement<m>:MAIN`. Other measurements are not considered.

See also: "Creating and reading histograms" on page 1263.

The following table lists the <MeasType> parameter values with a short description.

In addition, the suffixes <n> for limit-check remote commands are listed.

For a detailed description, see Table 8-8.

Table 24-10: Histogram measurements

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result	Suffix <n>
WCOunt	Waveform count	Number of acquisitions (waveforms) the histogram is based on	1
WSAMples	Waveform samples	Number of samples from the most recent acquisition included in the current histogram	2

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result	Suffix <n>
HSAMples	Histogram samples	Number of samples from all acquisitions included in the current histogram	3
HPEak	Histogram peak	Maximum sample value in the histogram	4
PEAK	Peak value	Signal value at the histogram peak	5
UPEakvalue	Upper peak value	Signal value at the maximum sample value in the upper half of the histogram	6
LPEakvalue	Lower peak value	Signal value at the maximum sample value in the lower half of the histogram	7
HMAXimum	Maximum	Highest signal value with a probability > 0	8
HMINimum	Minimum	Lowest signal value with a probability > 0	9
MEDian	Median	Signal value for which half the samples lie above, the other half below in the histogram	10
MAXMin	Max - Min	Range of signal values with a probability > 0	11
HMEan	Mean	Weighted arithmetic average of the histogram	12
HSTDdev	σ (S-dev)	Standard deviation of the sample numbers	13
M1STddev	Mean ±σ	Range between (mean value + standard deviation) and (mean value - standard deviation)	14
M2STddev	Mean ±2*σ	Range between (mean value + 3 * standard deviation) and (mean value - 2 * standard deviation)	15
M3STddev	Mean ±3*σ	Range between (mean value + 3 * standard deviation) and (mean value - 2 * standard deviation)	16
MKPositive	Marker + Probability %	Marker value (according to the selected probability domain marker type) plus the defined limit. Note that the value is restricted to the histogram range.	17
MKNegative	Marker - Probability %	Marker value (according to the selected probability domain marker type) minus the defined limit. Note that the value is restricted to the histogram range.	18

MEASurement<m>:HISTogram:SELect 1528
MEASurement<m>:HISTogram:PROBability:TYPE 1529
MEASurement<m>:HISTogram:PROBability:LIMit 1529
MEASurement<m>:HISTogram:LCHeck<n>:VALid 1530
MEASurement<m>:HISTogram:LCHeck<n>:LOWer:LIMit 1530
MEASurement<m>:HISTogram:LCHeck<n>:UPPer:LIMit 1530
MEASurement<m>:HISTogram:LCHeck<n>:LOWer:MARGin 1531
MEASurement<m>:HISTogram:LCHeck<n>:UPPer:MARGin 1531

MEASurement<m>:HISTogram:SELect <HistgRef>

Selects the histogram on which the measurement is based.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<HistgRef> String with the name of the histogram

Example: See "Creating and reading histograms" on page 1263

Usage: Asynchronous command

MEASurement<m>:HISTogram:PROBability:TYPE <HistgPrbDomMark>

Defines the marker reference in the probability domain.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<HistgPrbDomMark> PEAK | UPPK | LWPK | MAXimum | MINimum | MEDian | MEAN

PEAK
The y-value with the maximum sample value in the histogram

UPPK
The y-value at the maximum sample value in the upper half of the histogram

LWPK
The y-value at the maximum sample value in the lower half of the histogram

MAXimum
The highest y-value with a probability > 0

MINimum
The lowest y-value with a probability > 0

MEDian
The y-value for which half the samples lie above, the other half below in the histogram

MEAN
The weighted arithmetic average of the histogram

*RST: PEAK

Usage: Asynchronous command

MEASurement<m>:HISTogram:PROBability:LIMit <HistgPrbDomLim>

Defines a range around the probability marker.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<HistgPrbDomLim> Range: 0 to 100
Increment: 10
*RST: 10
Default unit: %

Usage: Asynchronous command

MEASurement<m>:HISTogram:LCHeck<n>:VALid <ValidRange>

Enables or disables limit checking for histogram measurements in the specified measurement channel.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..18
Number of the histogram measurement, see Table 24-10.

Parameters:

<ValidRange> ILIMit | ULIMit | LLIMit | OLIMit

ILIMit
Inside (within) limit; between the upper and lower limit values

ULIMit
Upper limit; above the upper limit value

LLIMit
Lower limit; below the lower limit value

OLIMit
Outside limit; above the upper limit or below the lower limit values

*RST: ILIMit

Usage: Asynchronous command

MEASurement<m>:HISTogram:LCHeck<n>:LOWer:LIMit <LowerLimit>
MEASurement<m>:HISTogram:LCHeck<n>:UPPer:LIMit <UpperLimit>

Define the lower and upper limits for the limit check, respectively. The valid range is defined using the `MEASurement<m>:HISTogram:LCHeck<n>:VALid` command.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..18
Number of the histogram measurement, see Table 24-10.

Parameters:
<UpperLimit> Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

MEASurement<m>:HISTogram:LCHeck<n>:LOWer:MARGin <LowerMargin>
MEASurement<m>:HISTogram:LCHeck<n>:UPPer:MARGin <UpperMargin>

Define the lower and upper margins for the limit check, respectively. Margins are not as strict as limits and must be within the valid value range. The valid range is defined using the `MEASurement<m>:HISTogram:LCHeck<n>:VALid` command.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..18
Number of the histogram measurement, see Table 24-10.

Parameters:
<UpperMargin> Range: -100 to 100
Increment: 10E-6
*RST: 0

Usage: Asynchronous command

24.12.9 Display

MEASurement<m>:DISPlay:LEVels 1531
MEASurement<m>:DISPlay:RESults 1532
MEASurement<m>:DISPlay:STYLe 1532
MEASurement<m>:DISPlay:HISTogram 1532

MEASurement<m>:DISPlay:LEVels <DisplayLevels>

If enabled, the reference levels used for the measurement are displayed in the diagram.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<DisplayLevels> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:DISPlay:RESults <DisplayResult>

If enabled, the intermediate result linesare displayed in the measurement diagram.

Result lines mark the samples in the waveform that are used to obtain the measurement result. These are, for example, maximum and minimum values, mean, cycle start and cycle end, and others.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<DisplayResult> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:DISPlay:STYLe <DisplayStyle>

Selects the style in which the measurement waveform is displayed.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<DisplayStyle> LINE | MARKer

LINE
The individual data points are connected by a line.

MARKer
Only the individual data points are displayed as markers.

*RST: LINE

Usage: Asynchronous command

MEASurement<m>:DISPlay:HISTogram <DispHistg>

Displays a histogram for the source of the selected measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<DispHistg> ON | OFF
*RST: OFF

Usage: Asynchronous command

24.12.10 Statistics and long-term measurements

See also: Chapter 24.16.7, "Long term measurement results and measurement histogram export to file", on page 1659.

MEASurement<m>:STATistics[:ENABle] 1533
MEASurement<m>:CLEar 1533
MEASurement<m>:MULTiple 1534
MEASurement<m>:MNOMeas 1534
MEASurement<m>:STATistics:HISTogram 1534
MEASurement<m>:STATistics:HBINs 1535
MEASurement<m>:STATistics:MODE 1535
MEASurement<m>:STATistics:RCOunt 1535
MEASurement<m>:STATistics:RMEascount 1536
MEASurement<m>:STATistics:RTIMe 1536
MEASurement<m>:STATistics:RESet 1536
MEASurement<m>:VERTical:CONT 1537
MEASurement<m>:VERTical:AUTO 1537
MEASurement<m>:VERTical:OFFSet 1537
MEASurement<m>:VERTical:SCALe 1538
MEASurement<m>:LTMeas[:STATe] 1538
MEASurement<m>:LTMeas:COUNt 1538
MEASurement<m>:LTMeas:TIME 1539

MEASurement<m>:STATistics[:ENABle] <StatisticsState>

Enables statistics calculation for the measurement.

For details on the statistics results, see Chapter 8.2.12.1, "Statistics", on page 356.

Suffix:
<m> See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<StatisticsState> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:CLEar

Deletes the statistic results of the indicated mesurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Usage: Event
Asynchronous command

MEASurement<m>:MULTiple <MultiMeas>

The measurement is performed repeatedly if the measured parameter occurs several times inside the acquisition or defined gate.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<MultiMeas> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:MNOMeas <MaxMeasPerAcq>

Sets the maximum number of measurements per acquisition if repeated measurement is enabled (`MEASurement<m>:MULTiple` is `ON`).

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<MaxMeasPerAcq> Range: 2 to 1000000
Increment: 1
*RST: 1000

Usage: Asynchronous command

MEASurement<m>:STATistics:HISTogram <ShowHistogram>

Displays a histogram of the statistical results. Enabling the histogram enables also the calculation and display of statistics for the measurement results if statistics were disabled. the histogram shows the cumulative occurence distribution of mean measurement results in a graphic.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<ShowHistogram> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:STATistics:HBINs <HistogramBins>

Sets the number of bins - the number of vertical bars that build the histogram.

If `MEASurement<m>:VERTical:CONT` is ON, the instrument determines the number of bins automatically based on the time base, the current measurements, and other settings.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<HistogramBins> Range: 2 to 1000
Increment: 10
*RST: 1000

Usage: Asynchronous command

MEASurement<m>:STATistics:MODE <ResetMode>

Defines when the statistics for long term measurements are reset.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<ResetMode> TIME | WFMS | MEAS

TIME
Sets one long term measurement point after the time defined using `MEASurement<m>:STATistics:RTIMe`.

WFMS - Waveforms
Sets one long term measurement point after a number of acquired waveforms defined using `MEASurement<m>:STATistics:RCOunt`.

MEAS
Sets one long term measurement point after a number of measurement results.

*RST: TIME

Usage: Asynchronous command

MEASurement<m>:STATistics:RCOunt <RstWfmCnt>

Defines the number of measured waveforms from which one point of the long term measurement is created (reset of statistics).

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<RstWfmCnt> Range: 1 to 65535
Increment: 10
*RST: 1000

Usage: Asynchronous command

MEASurement<m>:STATistics:RMEascount <RstMeasCnt>

Defines the number of measurement results from which one point of the long term measurement is created.

This setting is only available if `MEASurement<m>:STATistics:MODE` is set to `MEAS`.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<RstMeasCnt> Range: 1 to 65535
Increment: 10
*RST: 1000

Usage: Asynchronous command

MEASurement<m>:STATistics:RTIMe <ResetTime>

Defines the time or period after which the statistics are reset.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<ResetTime> Range: 0.1 to 2.14748E+9
Increment: 1E-3
*RST: 0.2
Default unit: s

Usage: Asynchronous command

MEASurement<m>:STATistics:RESet

Resets the histogram, the long term measurement and the statistics.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Usage: Event
Asynchronous command

MEASurement<m>:VERTical:CONT <ContAutoScl>

If enabled, automatic vertical scaling is performed whenever the waveform does not fit in the diagram during the long term measurement period.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<ContAutoScl> ON | OFF
*RST: ON

Usage: Asynchronous command

MEASurement<m>:VERTical:AUTO

Performs an automatic scaling once so that the scaling is adapted to the current measurement results. Available only for long term measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Usage: Event
Asynchronous command

MEASurement<m>:VERTical:OFFSet <VerticalOffset>

Defines a vertical offset for the long term measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<VerticalOffset> Range: -100E+12 to 100E+12
Increment: 1E-6
*RST: 0
Default unit: div

Usage: Asynchronous command

MEASurement<m>:VERTical:SCALe <VerticalScale>

Defines the vertical scaling per division, so that the scaling can be adapted automatically during the long term measurement period.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<VerticalScale> Range: 1E-12 to 100E+12
Increment: 10E-6
*RST: 0.5
Default unit: V/div

Usage: Asynchronous command

MEASurement<m>:LTMeas[:STATe] <ShowDiagram>

Enables long term measurement for a defined number of measurement points (see `MEASurement<m>:LTMeas:COUNt` on page 1538) or a specified time (see`MEASurement<m>:LTMeas:TIME` on page 1539).

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<ShowDiagram> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:LTMeas:COUNt <MeasCnt>

Defines the total number of points to be measured during the long term measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<MeasCnt> Range: 1000 to 200000
Increment: 10
*RST: 1000

Usage: Asynchronous command

MEASurement<m>:LTMeas:TIME <MeasurementTime>

Defines the total duration of the long term measurement.

This setting is only available if `MEASurement<m>:STATistics:MODE` is set to "Time".

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<MeasurementTime> Range: 0.01 to 2.14748E+9
Increment: 1
*RST: 200
Default unit: s

Usage: Asynchronous command

24.12.11 Protocol

Before starting the protocol measurement, a bus type has to be selected (`BUS<m>:TYPE`) and the bus has to be activated (`BUS<m>[:STATe]`).

The following table lists the <MeasType> parameter values with a short description.

Table 24-11: Protocol measurements

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result
AUDio	Audio	
FLDValue	Field value	Value of a field over time
F2F	Frame to frame	The distance between two frame types
	Trigger to frame	
GAP	Gap	Measures a gap, periods at which the bus is idle. The distance of a gap can only be measured between two identified frames.
MBITrate	Main bit rate	Bit rate of the protocol
SBITrate	2nd bit rate	Additional bit rate, protocol dependent.
BIDLe	Bus idle	Calculates the bus idle time
FCNT	Frame count	Number of all frames within the acquisition window
FEC	Frame error count	Sum of all frames with errors within the acquisition window

<MeasType> parameter value (remote control)	Measurement (manual operation)	Description/Result
FER	Frame error rate	Sum of all frames with errors divided by all frames within the acquisition window
CFER	Consecutive frame error rate	Measures the frame error rate

MEASurement<m>:PROTocol:F2FRame:FLDFrom 1540
MEASurement<m>:PROTocol:F2FRame:FLDTo 1540
MEASurement<m>:PROTocol:F2FRame:FRMFrom 1541
MEASurement<m>:PROTocol:F2FRame:FRMTo 1541
MEASurement<m>:PROTocol:F2FRame:VALFrom 1541
MEASurement<m>:PROTocol:F2FRame:VALTo 1541
MEASurement<m>:PROTocol:FLDValue:FLD 1542
MEASurement<m>:PROTocol:FLDValue:FRM 1542
MEASurement<m>:PROTocol:FLDValue:TRCK 1542
MEASurement<m>:PROTocol:FLDValue:VAL 1542
MEASurement<m>:PROTocol:MBITrate:FLD 1543
MEASurement<m>:PROTocol:MBITrate:FRM 1543
MEASurement<m>:PROTocol:MBITrate:VAL 1543
MEASurement<m>:PROTocol:SBITrate:FLD 1543
MEASurement<m>:PROTocol:SBITrate:FRM 1543
MEASurement<m>:PROTocol:SBITrate:VAL 1544
MEASurement<m>:PROTocol:T2FRame:DIRection 1544
MEASurement<m>:PROTocol:T2FRame:FLD 1544
MEASurement<m>:PROTocol:T2FRame:FRM 1544
MEASurement<m>:PROTocol:T2FRame:VALue 1545

MEASurement<m>:PROTocol:F2FRame:FLDFrom <FieldName>

Sets the type of field for the start frame of the frame to frame protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<FieldName>

MEASurement<m>:PROTocol:F2FRame:FLDTo <FieldName>

Sets the type of field for the end frame of the frame to frame protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<FieldName>

MEASurement<m>:PROTocol:F2FRame:FRMFrom <FrameName>

Sets the type of start frame for the frame to frame protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<FrameName>

MEASurement<m>:PROTocol:F2FRame:FRMTo <FrameName>

Sets the type of end frame for the frame to frame protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<FrameName>

MEASurement<m>:PROTocol:F2FRame:VALFrom <FrmFrmFromVal>

Sets the field value of the start frame for the frame to frame protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<FrmFrmFromVal>

Usage: Asynchronous command

MEASurement<m>:PROTocol:F2FRame:VALTo <FrmFrmVal>

Sets the field value of the end frame for the frame to frame protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<FrmFrmVal>

Usage: Asynchronous command

MEASurement<m>:PROTocol:FLDValue:FLD <FieldName>

Sets the type of field for the frame of the field value protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<FieldName>

MEASurement<m>:PROTocol:FLDValue:FRM <FrameName>

Sets the type frame for the field value protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<FrameName>

MEASurement<m>:PROTocol:FLDValue:TRCK <FieldName>

Sets the field to be tracked value of the frame for the field value protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<FieldName>

MEASurement<m>:PROTocol:FLDValue:VAL <FieldValIdVal>

Sets the field value of the frame for the field value protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<FieldValIdVal>

Usage: Asynchronous command

MEASurement<m>:PROTocol:MBITrate:FLD <FieldName>

Sets the type of field for the frame of the main bit rate protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<FieldName>

MEASurement<m>:PROTocol:MBITrate:FRM <FrameName>

Sets the type frame for the main bit rate protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<FrameName>

MEASurement<m>:PROTocol:MBITrate:VAL <MainBtRateIdVal>

Sets the field value of the frame for the main bit rate protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<MainBtRateIdVal>

Usage: Asynchronous command

MEASurement<m>:PROTocol:SBITrate:FLD <FieldName>

Sets the type of field for the frame of the secondary bit rate protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<FieldName>

MEASurement<m>:PROTocol:SBITrate:FRM <FrameName>

Sets the type frame for the secondary bit rate protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<FrameName>

MEASurement<m>:PROTocol:SBITrate:VAL <SecondaryBtRateIdVal>

Sets the field value of the frame for the secondary bit rate protocol measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<SecondaryBtRateIdVal>

Usage: Asynchronous command

MEASurement<m>:PROTocol:T2FRame:DIRection <TrigFrmDirct>

Sets the direction of the field for trigger to frame protocol measurement.

Suffix:
<m> 1..10

Parameters:
<TrigFrmDirct> T2F | F2T

`T2F`: trigger to frame
`F2T`: frame to trigger

*RST: T2F

Usage: Asynchronous command

MEASurement<m>:PROTocol:T2FRame:FLD <FieldName>

Sets the type of the field for trigger to frame protocol measurement.

Suffix:
<m> 1..10

Parameters:
<FieldName>

MEASurement<m>:PROTocol:T2FRame:FRM <FrameName>

Sets the value of the frame for trigger to frame protocol measurement.

Suffix:
<m> 1..10

Parameters:
<FrameName>

MEASurement<m>:PROTocol:T2FRame:VALue <TrigFrmVal>

Sets the value of the field for trigger to frame protocol measurement.

Suffix:
<m> 1..10

Parameters:
<TrigFrmVal>

Usage: Asynchronous command

24.12.12 Track and trend

MEASurement<m>:TRACk[:STATe]..... 1545
MEASurement<m>:TRACk:DATA:HEADer?.....1545
MEASurement<m>:TRACk:DATA:STYPe?..... 1546
MEASurement<m>:TRACk:DATA[:VALues]?.....1546

MEASurement<m>:TRACk[:STATe] <State>

Enables the track functionality and displays the track.

The track functionality requires at least one option, see "Enable (Track)" on page 361.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:TRACk:DATA:HEADer?

Returns the header of the track.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Usage: Query only

MEASurement<m>:TRACk:DATA:STYPe?

Returns the data type: TRK (track).

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Usage: Query only

MEASurement<m>:TRACk:DATA[:VALues]?

Returns the data of track points for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

To set the export format, use `FORMat[:DATA]`.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Usage: Query only

24.12.13 Gating

MEASurement<m>:GATE[:STATe] 1546
MEASurement<m>:GATE:MODE 1547
MEASurement<m>:GATE:ABSolute:STARt 1547
MEASurement<m>:GATE:ABSolute:STOP 1547
MEASurement<m>:GATE:RELative:STARt 1547
MEASurement<m>:GATE:RELative:STOP 1547
MEASurement<m>:GATE:CURSor 1548
MEASurement<m>:GATE:CCOupling 1548
MEASurement<m>:GATE:ZCOupling 1548
MEASurement<m>:GATE:ZDIagram 1549
MEASurement<m>:GATE:GCOupling 1549

MEASurement<m>:GATE[:STATe] <State>

Considers the gating settings of the source waveform for the measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:GATE:MODE <Mode>

Defines whether the gate settings are configured using absolute or relative values.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<Mode> ABS | REL
*RST: ABS

Usage: Asynchronous command

MEASurement<m>:GATE:ABSolute:STARt <Start>
MEASurement<m>:GATE:ABSolute:STOP <Stop>

Define the absolute start and end values for the gate.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<Stop> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Usage: Asynchronous command

MEASurement<m>:GATE:RELative:STARt <RelativeStart>
MEASurement<m>:GATE:RELative:STOP <RelativeStop>

Define the relative start and end values for the gate, respectively.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<RelativeStop> Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Usage: Asynchronous command

MEASurement<m>:GATE:CURSor <Cursorset>

Selects the cursor set to be used for measurement gating. The gate area is defined by the cursor lines.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<Cursorset> CURSOR1 | CURSor1 | CURSOR2 | CURSor2 | CURSOR3 | CURSor3 | CURSOR4 | CURSor4

CURSOR1 = CURSor1, CURSOR2 = CURSor2, CURSOR3 = CURSor3, CURSOR4 = CURSor4

*RST: CURSOR1

Usage: Asynchronous command

MEASurement<m>:GATE:CCOupling <CursorCoupling>

Enables the cursor coupling for automatic measurements.

Select the cursor set to be used with `MEASurement<m>:GATE:CURSor`.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<CursorCoupling> ON | OFF

*RST: OFF

Usage: Asynchronous command

MEASurement<m>:GATE:ZCOupling <ZoomCoupling>

If enabled, the gate area is defined identically to the zoom area for the zoom diagram.

If enabled, define the zoom area to be used as gate with `MEASurement<m>:GATE:ZDIagram`.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<ZoomCoupling> ON | OFF

*RST: OFF

Usage: Asynchronous command

MEASurement<m>:GATE:ZDIagram <DiagramName>, <ZoomName>

If `MEASurement<m>:GATE:ZCOupling` is enabled, the gate area is defined identically to the zoom area for the selected zoom diagram.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<ZoomDiagram> String with the name of the diagram on which the zoom is based

Example: Prerequisite: Zoom 1 is available
```
MEASurement:GATE:ZCOupling ON
MEASurement:GATE:ZDIagram "Diagram1","Zoom1"
```

Usage: Asynchronous command

MEASurement<m>:GATE:GCOupling <GateCoupling>

If you enable the gate coupling, the gate settings of the selected measurement are copied to all other measurements. If zoom or cursor coupling is active in a measurement, the zoom size and cursor positions are adjusted.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<GateCoupling> ON | OFF
*RST: OFF

Usage: Asynchronous command

24.12.14 Limit check and event actions

MEASurement<m>:LCHeck 1549
MEASurement<m>:ONViolation:BEEP 1550
MEASurement<m>:ONViolation:ACQStop 1550
MEASurement<m>:ONViolation:WFMSave 1551
MEASurement<m>:ONViolation:REPort 1551
MEASurement<m>:ONViolation:TRIGgerout 1551
MEASurement<m>:ONViolation:RUNexec 1551

MEASurement<m>:LCHeck <LimitCheckState>

Defines the type of the limit check that can run together with the measurement.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<LimitCheckState> OFF | LIMit | LMARgin

OFF
No limit check.

LIMit
Only limits are checked.

LMARgin
Limits and margins are checked.

*RST: OFF

Usage: Asynchronous command

MEASurement<m>:ONViolation:BEEP <Beep>

Generates a beep sound for the specified event.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<Beep> NOACtion | SUCCess | VIOLation

See Chapter 24.4.5, "Event parameter", on page 1286

*RST: NOACtion

Usage: Asynchronous command

MEASurement<m>:ONViolation:ACQStop <StopAcq>

Stops data acquisition for the specified event.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<StopAcq> NOACtion | SUCCess | VIOLation

See Chapter 24.4.5, "Event parameter", on page 1286

*RST: NOACtion

Usage: Asynchronous command

MEASurement<m>:ONViolation:WFMSave <SaveWfm>

Saves the waveform data.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<SaveWfm> NOACtion | SUCCess | VIOLation
See Chapter 24.4.5, "Event parameter", on page 1286
*RST: NOACtion

Usage: Asynchronous command

MEASurement<m>:ONViolation:REPort <Report>

Creates and saves a report of the current settings and results.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<Report> NOACtion | SUCCess | VIOLation
See Chapter 24.4.5, "Event parameter", on page 1286
*RST: NOACtion

Usage: Asynchronous command

MEASurement<m>:ONViolation:TRIGgerout <TriggerOutPulse>

Creates a trigger out pulse on limit or margin violation or on successful completion of the measurement.

Suffix:
<m> 1..10

Parameters:
<TriggerOutPulse> NOACtion | SUCCess | VIOLation
*RST: NOACtion

Usage: Asynchronous command

MEASurement<m>:ONViolation:RUNexec <RunExecutable>

Starts an external application if the command is set to `VIOLation` or `SUCCess`.

Use the following commands to set up the application:

- `EXECutable:NAME` on page 1668

- EXECutable:PARameter on page 1669
- EXECutable:WDIRectory on page 1669

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<RunExecutable> NOACtion | SUCCess | VIOLation
See Chapter 24.4.5, "Event parameter", on page 1286
*RST: NOACtion

Usage: Asynchronous command

24.12.15 Reference levels

- General reference level settings 1552
- Automatic configuration 1554
- Manual configuration 1556
- Hysteresis 1561
- Tube 1561
- Results 1563

24.12.15.1 General reference level settings

REFLevel<m>:LDETection 1552
REFLevel<m>:LMODe 1553
REFLevel<m>:RELative:MODE 1553
REFLevel<m>:USRLevel 1554

REFLevel<m>:LDETection <Mode>

Defines whether the reference level is configured manually or automatically.

For automatic configuration, select the signal level to be used (see REFLevel<m>:AUTO:MODE on page 1554).

Suffix:

<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:

<Mode> AUTO | MANual

Example: REFLevel2:LDETection MANual
Sets manual level configuration for C1W1. C1W1 corresponds to suffix number 2.

Example: See: Chapter 24.3.2.2, "Setting reference levels", on page 1262

Usage: Asynchronous command

REFLevel<m>:LMODe <Mode>

Defines whether the reference is configured using absolute or relative values.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Mode> ABS | REL

Example: `REFLevel2:LMODe ABS`
Sets definition of reference levels to absolute values for C1W1. C1W1 corresponds to suffix number 2.

Example: See: "Manual reference level definition using relative values" on page 1262

Usage: Asynchronous command

REFLevel<m>:RELative:MODE <Mode>

The lower, middle and upper reference levels, defined as percentages of the high signal level.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Mode> FIVE | TEN | TWENty | USER

FIVE
5/50/95

TEN
10/50/90

TWENty
20/50/80

USER
Set the reference levels to individual values with `REFLevel<m>:RELative:LOWer`, `REFLevel<m>:RELative:MIDDle`, and `REFLevel<m>:RELative:UPPer`.

Example: `REFL2:REL:MODE FIVE`
Reference levels for C1W1: Lower reference level = 5% of high signal level, middle reference level = 50% of high signal level, upper reference level = 95% of high signal level

Example: See: "Manual reference level definition using relative values" on page 1262

Usage: Asynchronous command

REFLevel<m>:USRLevel <Mode>

Defines whether the user-defined signal levels or user-defined reference levels are used for the measurements.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Mode> USIGnal | UREF

USIGnal
The high and low signal levels are defined by the user.

UREF
The reference levels are defined by the user.

Example: `REFLevel2:USRLevel UREF`
Sets user-defined reference levels to be used for C1W1. C1W1 corresponds to suffix number 2.

Example: See: "Manual reference level definition using absolute values" on page 1262

Usage: Asynchronous command

24.12.15.2 Automatic configuration

REFLevel<m>:AUTO:MODE 1554
REFLevel<m>:AUTO[:STATe] 1555
REFLevel<m>:AUTO:COUNt 1556

REFLevel<m>:AUTO:MODE <Mode>

Defines the high and low signal levels from which the reference levels are derived.

This setting is only available for automatic reference level mode (see `REFLevel<m>:LDETection` on page 1552).

Suffix:

<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:

<Mode> AUTO | PPRobability | MPRobability | ABSolutepeak | UPLM | UMLP | UALM | UMLA

AUTO
Auto select absolute probability: most suitable signal levels for the selected measurement

PPRobability
Peak probability: signal levels with the highest probability value

MPRobability
Mean probability: signal levels with mean probability

ABSolutepeak
Absolute peak: absolute peak signal levels

UPLM
Upper absolute peak, lower mean probability: high signal level is the upper absolute peak, low signal level is the level with the mean probability in the lower half of the histogram.

UMLP
Upper mean probability, lower absolute peak: high signal level is the level with mean probability in the upper half of the histogram, low signal level is the lower absolute peak.

UALM
Upper absolute peak, lower manual value: high signal level is the maximum result value of the amplitude measurement; low signal level is manually set using `REFLevel<m>:RELative:LOWer`.

UMLA
Upper manual, lower absolute peak: The upper signal level is set manually using `REFLevel<m>:RELative:UPPer`. The lower level is the minimum result value of the amplitude measurement.

Example: `REFLevel5:AUTO:MODE PPRobability`
Sets the automatic reference level mode for C2W1 to "Peak probability". C2W1 corresponds to suffix number 5.

Example: See: "Automatic level detection, peak probability" on page 1263

Usage: Asynchronous command

REFLevel<m>:AUTO[:STATe] <Value>

Enables averaging over several histograms to determine the reference levels. The number of histograms to consider is defined using `REFLevel<m>:AUTO:COUNt`.

This function is only available in automatic reference level mode (see `REFLevel<m>:LDETection` on page 1552).

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value> ON | OFF

Usage: Asynchronous command

REFLevel<m>:AUTO:COUNt <Value>

Defines the number of histograms from which the average is calculated.

Prerequisites:

- `REFLevel<m>:AUTO[:STATe]` is set to `ON`
- `REFLevel<m>:LDETection` on page 1552 is set to `AUTO`

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value>

Usage: Asynchronous command

24.12.15.3 Manual configuration

REFLevel<m>:ABSolute:HIGH 1556
REFLevel<m>:ABSolute:LOW 1557
REFLevel<m>:ABSolute:TDIStance 1557
REFLevel<m>:ABSolute:BDIStance 1558
REFLevel<m>:ABSolute:MLEVel 1558
REFLevel<m>:ABSolute:ULEVel 1558
REFLevel<m>:ABSolute:LLEVel 1559
REFLevel<m>:RELative:UPPer 1559
REFLevel<m>:RELative:MIDDle 1560
REFLevel<m>:RELative:LOWer 1560

REFLevel<m>:ABSolute:HIGH <Value>

The signal value that represents a high level.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value>

Example: `REFLevel2:ABSolute:HIGH 0.015`
Sets the high signal level for C1W1 to 15 mV. C1W1 corresponds to suffix number 2.

Usage: Asynchronous command

REFLevel<m>:ABSolute:LOW <Value>

The signal value that represents a low level.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value>

Example: `REFLevel2:ABSolute:Low 0.0015`
Sets the low signal level forC1W1 to 1.5 mV. C1W1 corresponds to suffix number 2.

Usage: Asynchronous command

REFLevel<m>:ABSolute:TDIStance <Value>

The distance between the high signal level and the upper reference level.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value>

Example: `REFLevel2:ABSolute:TDIStance 0.0002`
Sets the top distance for C1W1 to 0.2 mV. C1W1 corresponds to suffix number 2.

Example: See: "Manual reference level definition using absolute values" on page 1262

Usage: Asynchronous command

REFLevel<m>:ABSolute:BDIStance <Value>

The distance between the lower reference level and the low signal value.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value>

Example: `REFLevel2:ABSolute:BDIStance 0.0002`
Sets the bottom distance for C1W1 to 0.2 mV. C1W1 corresponds to suffix number 2.

Example: See: "Manual reference level definition using absolute values" on page 1262

Usage: Asynchronous command

REFLevel<m>:ABSolute:MLEVel <Value>

For user signal level selection, the level is the middle level between high and low signal level. The value is adjusted automatically if you change the high or low signal levels. Vice versa, if you change the middle level, the high and low signal levels are adjusted.

For user reference level selection, the level is the middle level between upper and lower reference level. The value is adjusted automatically if you change the upper or lower reference levels. Vice versa, if you change the middle level, the upper and lower reference levels are adjusted.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value>

Example: `REFLevel2:ABSolute:MLEVel 0.05`
Sets the middle signal level for C1W1 to 50 mV. C1W1 corresponds to suffix number 2.

Example: See: "Manual reference level definition using absolute values" on page 1262

Usage: Asynchronous command

REFLevel<m>:ABSolute:ULEVel <Value>

The upper reference level, required e.g. to determine a rise.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value>

Example: `REFLevel2:ABSolute:ULEVel 0.01`
Sets the upper reference level for C1W1 to 10 mV. C1W1 corresponds to suffix number 2.

Example: See: "Manual reference level definition using absolute values" on page 1262

Usage: Asynchronous command

REFLevel<m>:ABSolute:LLEVel <Value>

The lower reference level, required e.g. to determine a fall.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value>

Example: `REFLevel2:ABSolute:LLEVel 0.001`
Sets the lower reference level for C1W1 to 1 mV. C1W1 corresponds to suffix number 2.

Example: See: "Manual reference level definition using absolute values" on page 1262

Usage: Asynchronous command

REFLevel<m>:RELative:UPPer <Value>

Sets the upper relative reference level if `REFLevel<m>:RELative:MODE` is set to `USER`.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value> Percentage of the high signal level.

Example: `REFLevel8:RELative:LOWer 85`
Sets the upper reference level for C3W1 to 85 %. C3W1 corresponds to suffix number 8.

Example: See: "Manual reference level definition using relative values" on page 1262

Usage: Asynchronous command

REFLevel<m>:RELative:MIDDle <Value>

Sets the middle relative reference level if `REFLevel<m>:RELative:MODE` is set to `USER`.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value> Percentage of the high signal level.

Example: `REFLevel8:RELative:MIDDle 50`
Sets the middle reference level for C3W1 to 50 %. C3W1 corresponds to suffix number 8.

Example: See: "Manual reference level definition using relative values" on page 1262

Usage: Asynchronous command

REFLevel<m>:RELative:LOWer <Value>

Sets the lower relative reference level if `REFLevel<m>:RELative:MODE` is set to `USER`.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value> Percentage of the high signal level.

Example: `REFLevel8:RELative:LOWer 15`
Sets the lower reference level for C3W1 to 15 %. C3W1 corresponds to suffix number 8.

Example: See: "Manual reference level definition using relative values" on page 1262

Usage: Asynchronous command

24.12.15.4 Hysteresis

REFLevel<m>:RELative:HYSTeresis <Value>

Defines a hysteresis for the middle reference level. A rise or fall from the middle reference value that does not exceed the hysteresis is rejected as noise.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value>

Usage: Asynchronous command

24.12.15.5 Tube

REFLevel<m>:RELative:OTUBe....1561
REFLevel<m>:RELative:ITUBe....1561
REFLevel<m>:ABSolute:TOTube....1562
REFLevel<m>:ABSolute:TITube....1562
REFLevel<m>:ABSolute:BITube....1562
REFLevel<m>:ABSolute:BOTube....1563

REFLevel<m>:RELative:OTUBe <Value>

Defines a percentage of the signal level by which the absolute signal level may be larger than the high signal level or lower than the low signal level to be considered high or low, respectively.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value>

Usage: Asynchronous command

REFLevel<m>:RELative:ITUBe <Value>

Defines a percentage of the signal level by which the absolute signal level may be higher than the low signal level or lower than the high signal level to be considered low or high, respectively.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value>

Usage: Asynchronous command

REFLevel<m>:ABSolute:TOTube <Value>

Defines an area above the high signal level which is still considered to be high level.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value>

Usage: Asynchronous command

REFLevel<m>:ABSolute:TITube <Value>

Defines an area beneath the high signal level which is still considered to be high level.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value>

Usage: Asynchronous command

REFLevel<m>:ABSolute:BITube <Value>

Defines an area above the low signal level which is still considered to be low level.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value>

Usage: Asynchronous command

REFLevel<m>:ABSolute:BOTube <Value>

Defines an area beneath the low signal level which is still considered to be low level.

Suffix:
<m> Valid suffix numbers: 2 to 21, 61 to 68, 108 to 118, 151 to 157, 173 to 179
Source waveform of the measurement, see Chapter 24.4.1, "Waveform suffix", on page 1279.

Parameters:
<Value>

Usage: Asynchronous command

24.12.15.6 Results

MEASurement<m>:REFLevel:RESult:LOWer?....1563
MEASurement<m>:REFLevel:RESult:MIDDle?....1563
MEASurement<m>:REFLevel:RESult:UPPer?....1563
MEASurement<m>:REFLevel:RESult:SIGLow?....1563
MEASurement<m>:REFLevel:RESult:SIGHigh?....1563
MEASurement<m>:REFLevel:RESult:BINNer?....1564
MEASurement<m>:REFLevel:RESult:BOUTer?....1564
MEASurement<m>:REFLevel:RESult:TINNer?....1564
MEASurement<m>:REFLevel:RESult:TOUTer?....1565

MEASurement<m>:REFLevel:RESult:LOWer?
MEASurement<m>:REFLevel:RESult:MIDDle?
MEASurement<m>:REFLevel:RESult:UPPer?

Return the lower, middle, and upper reference level, respectively.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Return values:
<Upper> Range: -100E+24 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

MEASurement<m>:REFLevel:RESult:SIGLow?
MEASurement<m>:REFLevel:RESult:SIGHigh?

Return the signal value that represents a low or high level, respectively.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Return values:
<SignalHigh> Range: -100E+24 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

MEASurement<m>:REFLevel:RESult:BINNer?

Returns the area above the low signal level which is still considered to be low level.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Return values:
<BottomInner> Range: -100E+24 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

MEASurement<m>:REFLevel:RESult:BOUTer?

Returns the area beneath the low signal level which is still considered to be low level.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Return values:
<BottomOuter> Range: -100E+24 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

MEASurement<m>:REFLevel:RESult:TINNer?

Returns the area beneath the high signal level which is still considered to be high level.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Return values:

<TopInner> Range: -100E+24 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

MEASurement<m>:REFLevel:RESult:TOUTer?

Returns the area above the high signal level which is still considered to be high level.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Return values:

<TopOuter> Range: -100E+24 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

24.13 Spectrum analysis

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

24.13.1 Basic FFT

CALCulate:MATH<m>:FFT:LOGScale....1566
CALCulate:MATH<m>:FFT:STARt....1566
CALCulate:MATH<m>:FFT:STOP....1567
CALCulate:MATH<m>:FFT:CFRequency....1567
CALCulate:MATH<m>:FFT:FULLspan....1567
CALCulate:MATH<m>:FFT:SPAN....1568
CALCulate:MATH<m>:FFT:TYPE....1568
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:ADJusted?....1568
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:AUTO....1569

CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:RATio....1569
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution][:VALue]....1569
CALCulate:MATH<m>:FFT:WINDow:TYPE....1569
CALCulate:MATH<m>:FFT:FRAMe:ARIThmetics....1571
CALCulate:MATH<m>:FFT:FRAMe:COVerage?....1571
CALCulate:MATH<m>:FFT:FRAMe:MAXCount....1571
CALCulate:MATH<m>:FFT:FRAMe:OFACtor....1572
CALCulate:MATH<m>:FFT:GATE:COUPling....1572
TIMebase:RACTime?....1573
CALCulate:MATH<m>:FFT:GATE:ABSolute:STARt....1573
CALCulate:MATH<m>:FFT:GATE:ABSolute:STOP....1573
CALCulate:MATH<m>:FFT:GATE:MODE....1573
CALCulate:MATH<m>:FFT:GATE:RELative:STARt....1574
CALCulate:MATH<m>:FFT:GATE:RELative:STOP....1574
CALCulate:MATH<m>:FFT:GATE:ZCOupling....1574
CALCulate:MATH<m>:FFT:GATE[:STATe]....1575
CALCulate:MATH<m>:FFT:MAGNitude:LEVel....1575
CALCulate:MATH<m>:FFT:MAGNitude:RANGe....1575
CALCulate:MATH<m>:FFT:MAGNitude:SCALe....1575
CALCulate:MATH<m>:FFT:PHASe:SCALe....1577
CALCulate:MATH<m>:FFT:PHASe:SUPPression....1577
CALCulate:MATH<m>:FFT:PHASe:THReshold....1577
CALCulate:MATH<m>:FFT:PHASe:UNWRap....1577
CALCulate:MATH<m>:FFT:COUPled:WITH<1..8>....1578

CALCulate:MATH<m>:FFT:LOGScale <XAxisMode>

Defines the scaling method for the frequency (x-)axis of the spectrogram.

This command is only available if option R&S RTO6-K37 is installed.

Suffix:
<m> 1..8

Parameters:
<XAxisMode> LIN | LOG

LOG
Logarithmic scaling

LIN
Linear scaling

*RST: LIN

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:STARt <StartFreq>

Defines the start frequency of the displayed frequency span.

Suffix:
<m> 1..8
math waveform

Parameters:

<StartFreq> start frequency

Range: 0 to 5E+9
Increment: 1
*RST: 2E+9
Default unit: Hz

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:STOP <StopFreq>

Defines the stop frequency of the displayed frequency span.

Suffix:

<m> 1..8
math waveform

Parameters:

<StopFreq> stop frequency

Range: 0 to 5E+9
Increment: 1
*RST: 2E+9
Default unit: Hz

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:CFRequency <CenterFreq>

Defines the position of the displayed frequency range, which is (Center - Span/2) to (Center + Span/2). The width of the range is defined using the `CALCulate:MATH<m>:FFT:SPAN` command.

Suffix:

<m> 1..8
math waveform

Parameters:

<CenterFreq> center frequency

Range: 0 to 2E+12
Increment: 1
*RST: 2.5E+9
Default unit: Hz

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:FULLspan

Performs FFT calculation for the full frequency span.

Suffix:

<m> 1..8
math waveform

Usage: Event
Asynchronous command

CALCulate:MATH<m>:FFT:SPAN <FreqSpan>

The span is specified in Hertz and defines the width of the displayed frequency range, which is (Center - Span/2) to (Center + Span/2). The position of the span is defined using the `CALCulate:MATH<m>:FFT:CFRequency` command.

Suffix:
<m> 1..8
Math waveform

Parameters:
<FreqSpan> Frequency span

Range: 1 to 10E+9
Increment: 1
*RST: Bandwidth of the instrument model, bandwidth extension options are not considered
Default unit: Hz

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:TYPE <Type>

Selects between the magnitude and phase spectrum type.

Suffix:
<m> 1..8

Parameters:
<Type> MAGNitude | PHASe

*RST: MAGNitude

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:ADJusted?

Queries the effective resolution bandwidth.

Suffix:
<m> 1..8
Math waveform

Return values:
<AdjResBW> effective resolution bandwidth

Range: 0.01 to 20E+12
*RST: 0
Default unit: Hz

Usage: Query only

CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:AUTO <State>

Couples the frequency span to the RBW.

Suffix:
<m> 1..8
math waveform

Parameters:
<State> ON | OFF
*RST: ON

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:RATio <SpanRBWRatio>

This command defines the ratio of span (Hz) / resolution bandwidth (Hz).

Suffix:
<m> 1..8
math waveform

Parameters:
<SpanRBWRatio> ratio span / resolution bandwidth
Range: 1 to 1000
Increment: 1
*RST: 100

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:BANDwidth[:RESolution][:VALue] <ResolutionBW>

This command defines the resolution bandwidth.

Suffix:
<m> 1..8
math waveform

Parameters:
<ResolutionBW> resolution bandwidth
Range: 0.01 to 160E+6
Increment: 0.01
*RST: 2E+6
Default unit: Hz

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:WINDow:TYPE <WindowType>

Windowing helps minimize the discontinuities at the end of the measured signal interval and thus reduces the effect of spectral leakage, increasing the frequency resolution.

Various different window functions are provided in the R&S RTO6 to suit different input signals. Each of the window functions has specific characteristics, including some advantages and some trade-offs. These characteristics need to be considered carefully to find the optimum solution for the measurement task.

Suffix:

<m> 1..8
math waveform

Parameters:

<WindowType> RECTangular | HAMMing | HANN | BLACkharris | GAUSsian | FLATTOP2 | FLATtop2 | KAISerbessel

RECTangular
The rectangular window has the best frequency resolution, but a poor amplitude accuracy and is recommended for separating two tones with almost equal amplitudes and a small frequency distance.

HAMMing
The Hamming window is bell shaped and has a good frequency resolution and fair amplitude accuracy. It is recommended for frequency response measurements as well as sine waves, periodic signals and narrow-band noise

HANN
The Hann window is bell shaped and has a slightly worse frequency resolution but smaller sidelobe level than the Hamming window. The applications are the same.

BLACkharris
The Blackman window is bell shaped and has a poor frequency resolution, but very good amplitude accuracy. It is recommended mainly for signals with single frequencies to detect harmonics.

GAUSsian
Good frequiency resolution and best magnitude resolution, recommended for weak signals and short duration

FLATTOP2 = FLATtop2
The flattop window has a poor frequency resolution, but the best amplitude accuracy and the sharpest side lobe. It is recommended for accurate single-tone amplitude measurements.

KAISerbessel
The Kaiser-Bessel window has a fair frequency resolution and good amplitude accuracy, and a very high sidelobe level. It is recommended for separating two tones with differing amplitudes and a small frequency distance.

*RST: BLACkharris

Usage: Asynchronous command

Firmware/software: Version 3.35 and higher: Use FLATTOP2 or FLATtop2 instead of FLAT2

CALCulate:MATH<m>:FFT:FRAMe:ARIThmetics <Arithmetics>

The arithmetic mode defines how the final FFT result is calculated from the individual frame results.

Suffix:
<m> 1..8
Selects the math waveform.

Parameters:
<Arithmetics> OFF | ENVelope | AVERage | RMS | MINHold | MAXHold
See "Segment arithmetic" on page 383
*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:FRAMe:COVerage?

Due to the restriction of the number of frames (see `CALCulate:MATH<m>:FFT:FRAMe:MAXCount` on page 1571), the waveform may only be analyzed partially. This command queries the percentage of the trace that was analyzed, i.e. which part of the trace was included in the frame calculation.

Suffix:
<m> 1..8
math waveform

Return values:
<FrameCoverage> Range: 0 to 100
*RST: 100
Default unit: %

Usage: Query only
Asynchronous command

CALCulate:MATH<m>:FFT:FRAMe:MAXCount <MaxFrameCount>

Restricts the maximum number of frames to be calculated. Due to the other parameter settings, the required number of frames may become very high, thus slowing performance. By restricting the number of frames, you can avoid performance loss without changing the other parameters.

Suffix:
<m> 1..8
math waveform

Parameters:
<MaxFrameCount> Range: 1 to 10000
Increment: 10
*RST: 1000

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:FRAMe:OFACtor <OverlapFactor>

Defines the minimum factor by which two neighboring frames overlap. If the required number of frames to cover the input values allows for more overlap, the factor is increased.

The higher the overlap factor, the more frames are used. This leads to more individual results and improves detection of transient signal effects. However, it also extends the duration of the calculation.

Suffix:
<m> 1..8
math waveform

Parameters:
<OverlapFactor> Range: 0 to 90
Increment: 1
*RST: 50
Default unit: %

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:GATE:COUPling <GateRBWCoupling>

Defines the behaviour of the record length or RBW value in dependancy to the other FFT parameters.

See also:

- "Record Length/RBW Coupling" on page 386
- Chapter 9.1.1, "Fundamentals of FFT analysis", on page 370

Suffix:
<m> 1..8
math waveform

Parameters:
<GateRBWCoupling> LENGth | RBW

LENGth
The record length remains constant. If not enough samples are available for the selected RBW, the RBW will be decreased.

RBW
The RBW is not adapted, i.e. remains as defined by the user. The required acquisition time for this RBW is indicated. If necessary and possible, the record length is extended to acquire the required number of samples.

*RST: RBW

Usage: Asynchronous command

TIMebase:RACTime?

Queries the required acquisition time. If FFT gating is used and the resolution BW is set to constant, record length can be extended to acquire the required number of samples. In this case, the required acquisition time differs from the adjusted acquisition time (`TIMebase:RANGe`).

Return values:

<RqrdAcqTime> Required acquisition time for FFT

Range: 125E-12 to 100E+3
*RST: 0.5
Default unit: s

Usage: Query only
Asynchronous command

CALCulate:MATH<m>:FFT:GATE:ABSolute:STARt <Start>

Defines the starting value for the gate.

Suffix:

<m> 1..8
math waveform

Parameters:

<Start> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:GATE:ABSolute:STOP <Stop>

Defines the end value for the gate.

Suffix:

<m> 1..8
math waveform

Parameters:

<Stop> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:GATE:MODE <Mode>

Defines whether the gate settings are configured using absolute or relative values.

Suffix:

<m> 1..8
math waveform

Parameters:
<Mode> ABS | REL
*RST: ABS

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:GATE:RELative:STARt <RelativeStart>

Defines the starting value for the gate in percent.

Suffix:
<m> 1..8
math waveform

Parameters:
<RelativeStart> Range: 0 to 100
Increment: 0.1
*RST: 0
Default unit: %

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:GATE:RELative:STOP <RelativeStop>

Defines the end value for the gate in percent.

Suffix:
<m> 1..8
math waveform

Parameters:
<RelativeStop> Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:GATE:ZCOupling <ZoomCoupling>

If enabled, the gate area is defined identically to the zoom area for the zoom diagram.

Suffix:
<m> 1..8
math waveform

Parameters:
<ZoomCoupling> ON | OFF
*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:GATE[:STATe] <State>

Enables FFT gating.

Suffix:
<m> 1..8
math waveform

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:MAGNitude:LEVel <VerticalMax>

Defines the reference level for dB scaling.

Suffix:
<m> 1..8
Math waveform

Parameters:
<VerticalMax> Range: -1E+15 to 1E+15
Increment: 0.01
*RST: 0
Default unit: div

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:MAGNitude:RANGe <Range>

Defines the vertical value range in spectrum mode.

Suffix:
<m> 1..8
math waveform

Parameters:
<Range> Range: 1 to 500
Increment: 1
*RST: 100
Default unit: dB

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:MAGNitude:SCALe <MagnitudeScale>

Defines the scaling of the y-axis for magnitude spectrums. The display values are valid for 50Ω termination impendance.

Select the unit according to the vertical unit of the source waveform.

Suffix:

<m> 1..8
math waveform

Parameters:

<MagnitudeScale> LINear | DBM | DB | DBUV | DBMV | DBV | DBPS | DBNS | DBUS | DBMS | DBS | DBHZ | DBKHZ | DBKHz | DBMHZ | DBMHz | DBGHZ | DBGHz | DBA | DBMA | DBUA

LINear
Linear scaling; displays the RMS value.

DBM | DB | DBUV | DBMV | DBV
Basic FFT

DBPS | DBNS | DBUS | DBMS | DBS | DBHZ | DBKHZ | DBKHz | DBMHZ | DBMHz | DBGHZ | DBGHz
FFT on tracks of time or frequency measurements

DBA | DBMA | DBUA
FFT on current waveforms

*RST: DBM

Usage: Asynchronous command

Table 24-12: Logarithmic scaling values

DBM	dBm
DB	dB (related to reference level) Set the reference level with `CALCulate:MATH<m>:FFT:MAGNitude:LEVel`.
DBUV	dBµV
DBMV	dBmV
DBV	dBV
DBPS	dBps
DBNS	dBns
DBUS	dBµs
DBMS	dBms
DBS	dBs
DBHZ	dBHz
DBKHZ = DBKHz	dBkHz
DBMHZ = DBMHz	dBMHz
DBGHZ = DBGHz	dBGHz
DBA	dBA
DBMA	dBmA
DBUA	dBµA

CALCulate:MATH<m>:FFT:PHASe:SCALe <PhaseScale>

Defines the scaling unit for phase display.

Suffix:
<m> 1..8
math waveform

Parameters:
<PhaseScale> DEGRees | RADians
*RST: DEGRees

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:PHASe:SUPPression <Suppression>

Enables noise suppression. Phase calculation is restricted to frequencies with a minimum magnitude, the threshold value (see `CALCulate:MATH<m>:FFT:PHASe:THReshold` on page 1577).

Suffix:
<m> 1..8
math waveform

Parameters:
<Suppression> ON | OFF
*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:PHASe:THReshold <SupprThres>

Defines the minimum frequency magnitude for which phases are calculated. This setting is only available if `CALCulate:MATH<m>:FFT:PHASe:SUPPression` is set to "ON".

Suffix:
<m> 1..8
math waveform

Parameters:
<SupprThres> Range: -180 to 180
Increment: 0.1
*RST: 0
Default unit: dBm

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:PHASe:UNWRap <Unwrap>

If enabled, phase shifts due to a limitation of the value range are eliminated.

Suffix:
<m> 1..8
math waveform

Parameters:
<Unwrap> ON | OFF
*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:COUPled:WITH<1..8> <MathIndex>

Copies the current FFT settings of the selected math waveform (m) to the other selected math waveform (m2), and couples the two waveforms. This can be repeated for all math waveforms.

If any FFT setting for any of the coupled spectrums is changed, it is changed for all coupled spectrums.

Suffix:
<m> 1..8
FFT waveforms to be coupled. <m>, <m2> must be active and distinct.

Parameters:
<MathIndex> ON | OFF
*RST: OFF

Example: Prerequisite: FFt waveforms M1 and M2 are active.
`CALC:MATH1:FFT:COUP:WITH2 ON`
Couples the math waveforms M1 and M2.

Usage: Asynchronous command

24.13.2 Waveform data

CALCulate:MATH<m>:DATA:STYPe? 1578
CALCulate:MATH<m>:DATA:HEADer? 1579
CALCulate:MATH<m>:DATA[:VALues]? 1579

CALCulate:MATH<m>:DATA:STYPe?

Returns the signal type of the source of the math waveform.

Suffix:
<m> 1..8
Selects the math waveform.

Return values:
<SignalType> SOUR | SPEC | CORR | MEAS | NONE
SOURce = normal signal
SPECtrum = FFT spectrum, specific math signal

CORRelation = correlated signal, specific math signal
MEAsurement = result of a measurement
NONE = undefined

Usage: Query only
Asynchronous command

CALCulate:MATH<m>:DATA:HEADer?

Returns the header of math waveform data. The header contains attributes of the waveform.

Table 24-13: Header data

Position	Meaning	Example
1	XStart in s	-9.477E-008 = - 94,77 ns
2	XStop in s	9.477E-008 = 94,77 ns
3	Record length of the waveform in Samples	200000
4	Number of values per sample interval. For most waveforms the result is 1, for peak detect and envelope waveforms it is 2. If the number is 2, the number of returned values is twice the number of samples (record length).	1

Suffix:
<m> 1..8
Selects the math waveform.

Example:
```
CALC:MATH1:DATA:HEAD?
-9.477E-008,9.477E-008,200000,1
```

Usage: Query only

CALCulate:MATH<m>:DATA[:VALues]?

Returns the data of the math waveform points for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

To set the export format, use `FORMat[:DATA]`.

You can retrieve only Y-values, or X- and Y-values. Use `EXPort:WAVeform:INCXvalues` to define this.

Suffix:
<m> 1..8
Selects the math waveform.

Return values:
<Data> List of values according to the format and content settings.

Usage: Query only

24.13.3 Spectrum analysis (option R&S RTO6-K37)

In all `CALC:MATH<m>:FFT` commands, the suffix <m> selects the math waveform.

In all `CALC:MATH<m>:FFT:SPEC:TIM` commands, the suffix <m> selects the timeline.

CALCulate:MATH<m>:FFT:SPECtrogram:CMODe 1580
CALCulate:MATH<m>:FFT:USEColtab 1580
CALCulate:MATH<m>:FFT:SPECtrogram:STATe 1581
CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:POSition 1581
CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:STATe 1581

CALCulate:MATH<m>:FFT:SPECtrogram:CMODe <ColorTableMode>

Selects the color table mode for the frequency analysis display.

Suffix:
<m> 1..8

Parameters:
<ColorTableMode> INCI | AMPL

INCI
("Incidence") The display color is set depending on the frequency of occurrence of a value.

AMPL
("Amplitude") In the spectrogram and the frequency analysis display, the color is used to indicate the magnitude of the FFT signal. The higher the amplitude the higher the color in the assigned color table.

*RST: INCI

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:USEColtab <UseColorTable>

If enabled, the selected waveform is displayed according to its assigned color table. For information on the available color tables, see Chapter 4.3.2, "Color tables", on page 76.

If this option is disabled, the preset color of the selected channel source is displayed, and the intensity of the specific signal color varies according to the cumulative occurrence of the values.

Suffix:
<m> 1..8

Parameters:
<UseColorTable> ON | OFF

*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:SPECtrogram:STATe <State>

Enables the spectrogram display for a math waveform.

Suffix:
<m> 1..8

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:POSition <Position>

Defines the position of one of the two possible time lines in a spectrogram. The time line must be enabled first, using the `CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:STATe` command.

Suffix:
<m> 1..8

<n> 1..2

Parameters:
<Position> The position of the time line is defined by the index of the data acquisition in the history. How many acquisitions are available depends on the history settings.
Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Asynchronous command

CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:STATe <State>

Enables one of two possible time lines in a spectrogram diagram. A time line marks a single waveform in the spectrogram, that is: the power vs frequency results for the data acquired at a specific time. After enabling a time line, the results for that time are displayed in the spectrum diagram.

The position of the time line is defined using the `CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:POSition` command.

Suffix:
<m> 1..8

<n> 1..2

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

24.14 Mask testing

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

- Mask test definition 1582
- Mask definition: user mask 1586
- Mask definition: waveform mask 1592
- Event actions 1594
- Results 1597
- Eye mask testing 1599

24.14.1 Mask test definition

MTESt:ADD 1582
MTESt:REMove 1583
MTESt[:STATe] 1583
MTESt:SHOW 1583
MTESt:RST 1583
MTESt:SOURce 1584
MTESt:CONDition 1584
MTESt:TOLerance 1584
MTESt:CTYPe 1585
MTESt:FILE:NAME 1585
MTESt:FILE:SAVE 1586
MTESt:FILE:OPEN 1586
MTESt:FILE:DELete 1586

MTESt:ADD <MaskTestName>

Creates a new mask test definition with the specified name.

Setting parameters:

<MaskTestName> String with the name of the mask test

Example: See Chapter 24.3.3.1, "Creating a user mask", on page 1265

Usage: Setting only
Asynchronous command

MTESt:REMove <MaskTestName>

Deletes the mask test definition with the specified name.

Setting parameters:
<MaskTestName> String with the name of the mask test

Usage: Setting only
Asynchronous command

MTESt[:STATe] <MaskTestName>,<State>
MTESt[:STATe]? <MaskTestName>

Activates and deactivates the mask test. If the acquisition is running, the test starts immediately. Otherwise, the test starts when acquisition is started.

The testing is stopped when acquisition is stopped, also due to the `MTESt:ONViolation:STOP` command, or if `MASK[:STATe]` is set to "OFF".

The coommand needs `*OPC` command synchronisation.

Parameters:
<State> ON | OFF
*RST: OFF

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

Example: `MTEST:STAT 'MyMask', ON; *OPC?`
See Chapter 24.3.3.1, "Creating a user mask", on page 1265

Usage: Asynchronous command

MTESt:SHOW <State>

Switches the display of all mask segments on or off.

Parameters:
<State> ON | OFF
*RST: ON

Usage: Asynchronous command

MTESt:RST

Clears all totals and results in all "Mask Test" result boxes.

Usage: Event
Asynchronous command

MTESt:SOURce <MaskTestName>,<Source>
MTESt:SOURce? <MaskTestName>

Selects the waveform to be tested against the mask.

Parameters:

<Source> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | XY1 | XY2 | XY3 | XY4 | SG1TL1 | SG1TL2 | SG2TL1 | SG2TL2 | SG3TL1 | SG3TL2 | SG4TL1 | SG4TL2 | SG5TL1 | SG5TL2 | SG6TL1 | SG6TL2 | SG7TL1 | SG7TL2 | SG8TL1 | SG8TL2 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

Waveform to be tested, see Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C1W1

Parameters for setting and query:

<MaskTestName> String containing the name of the mask test

Usage: Asynchronous command

MTESt:CONDition <MaskTestName>,<PassFailMode>
MTESt:CONDition? <MaskTestName>

Sets the first criteria for a failed test, the kind of hits to be considered for test evaluation. A test has failed if the number of sample hits or acquisition hits exceeds the limit defined by `MTESt:TOLerance`.

Parameters:

<PassFailMode> SAMPles | ACQuisitions

SAMPles
Considers the number of samples that hit the mask.

ACQuisitions
Considers the number of acquisitions that contain at least one sample hit. How many samples hit the mask in that acquisition is not relevant.

*RST: SAMPles

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:TOLerance <MaskTestName>,<TolViolCnt>
MTESt:TOLerance? <MaskTestName>

Sets the second criteria for a failed test, the number of tolerable sample hits or acquisition hits. Use `MTESt:CONDition` to define which hits are considered for test evaluation.

Parameters:

<TolViolCnt> Range: 0 to 4000000000
Increment: 1
*RST: 0

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:CTYPe <MaskTestName>,<DefinitionType>
MTESt:CTYPe? <MaskTestName>

Sets the method of mask definition.

Parameters:

<DefinitionType> USER | WFML | EYEMask | PROTocol

USER
The mask segments are created by entering the numerical x- and y-values of the mask points.
See: Chapter 24.14.2, "Mask definition: user mask", on page 1586

WFML
The mask is created from the envelope of an existing waveform.
See: Chapter 24.14.3, "Mask definition: waveform mask", on page 1592

EYEMask
Mask for eye diagram testing. See: Chapter 24.22.6, "Eye mask testing", on page 2519.
Requires jitter option R&S RTO6-K12

*RST: USER

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:FILE:NAME <MaskTestName>, <Path>
MTESt:FILE:NAME? <MaskTestName>

Specifies a file to save the mask test.

Parameters:

<Path> String containing path and file name, format .xml

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

MTESt:FILE:SAVE <MaskTestName>

Saves the specified mask test. It contains the mask definition, defined actions and fail conditions.

Setting parameters:
<MaskTestName> String with the name of the mask test

Usage: Setting only
Asynchronous command

MTESt:FILE:OPEN <MaskTestName>

Loads the specified mask test to the instrument.

Setting parameters:
<MaskTestName> String with the name of the mask test

Usage: Setting only
Asynchronous command

MTESt:FILE:DELete <MaskTestName>

Deletes the specified mask test.

Setting parameters:
<MaskTestName> String with the name of the mask test

Usage: Setting only
Asynchronous command

24.14.2 Mask definition: user mask

The chapter contains commands required for the definition of user masks - `MTESt:CTYPe` is set to `USER`.

Segment and point indices

In remote control, the numbering of segments and points starts from 0. But in manual operation, the numbering starts from 1.

MTESt:SEGMent:STATe 1587
MTESt:SEGMent:ADD 1587
MTESt:SEGMent:COUNt? 1587
MTESt:SEGMent:INSert 1587
MTESt:SEGMent:REMove 1588
MTESt:SEGMent:CLEar 1588
MTESt:SEGMent:REGion 1588
MTESt:SEGMent:POINt:ADD 1589
MTESt:SEGMent:POINt:INSert 1589
MTESt:SEGMent:POINt:REMove 1589

MTESt:SEGMent:POINt:COUNt?....1589
MTESt:SEGMent:POINt:X....1590
MTESt:SEGMent:POINt:Y....1590
MTESt:SEGMent:RESCale:RECalculate....1590
MTESt:SEGMent:RESCale:XFACtor....1591
MTESt:SEGMent:RESCale:YFACtor....1591
MTESt:SEGMent:RESCale:XOFFset....1591
MTESt:SEGMent:RESCale:YOFFset....1592

MTESt:SEGMent:STATe <MaskTestName>, <MaskSegmIdx>,<State>
MTESt:SEGMent:STATe? <MaskTestName>, <MaskSegmIdx>

Enables and disables the mask segment. Disabled segments are not considered by running tests.

Parameters:
<State> ON | OFF
*RST: ON

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Usage: Asynchronous command

MTESt:SEGMent:ADD <MaskTestName>

Creates a new segment in the mask definition.

Setting parameters:
<MaskTestName> String with the name of the mask test

Example: See Chapter 24.3.3.1, "Creating a user mask", on page 1265

Usage: Setting only
Asynchronous command

MTESt:SEGMent:COUNt? <MaskTestName>

Returns the number of segments in the mask definition

Query parameters:
<MaskTestName> String with the name of the mask test

Return values:
<Count> Number of segments

Usage: Query only

MTESt:SEGMent:INSert <MaskTestName>, <MaskSegmIdx>

Inserts a new segment before the specified index in the mask definition.

Setting parameters:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Usage: Setting only
Asynchronous command

MTESt:SEGMent:REMove <MaskTestName>, <MaskSegmIdx>

Removes the specified segment from the mask definition.

Setting parameters:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Usage: Setting only
Asynchronous command

MTESt:SEGMent:CLEar <MaskTestName>

Deletes all mask segments of the indicated mask test.

Setting parameters:

<MaskTestName> String with the name of the mask test

Usage: Setting only
Asynchronous command

MTESt:SEGMent:REGion <MaskTestName>, <MaskSegmIdx>,<Region>
MTESt:SEGMent:REGion? <MaskTestName>, <MaskSegmIdx>

Defines the region of the segment that builds the mask.

Parameters:

<Region> UPPer | LOWer | INNer

UPPer
the segment points are connected to a line, the display area above this line is the mask segment

LOWer
the segment points are connected to a line, the display area below this line is the mask segment

INNer
the segment points form a closed geometrical shape, which is the mask segment

*RST: INNer

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Example: See Chapter 24.3.3.1, "Creating a user mask", on page 1265

Usage: Asynchronous command

MTESt:SEGMent:POINt:ADD <MaskTestName>, <MaskSegmIdx>

Adds a new point to the segment definition.

Setting parameters:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Example: See Chapter 24.3.3.1, "Creating a user mask", on page 1265

Usage: Setting only
Asynchronous command

MTESt:SEGMent:POINt:INSert <MaskTestName>, <MaskSegmIdx>, <MaskSegmPtIdx>

Inserts a new point before the specified mask segment point.

Setting parameters:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

<MaskSegmPtIdx> Number of the point. Counting starts from 0.

Usage: Setting only
Asynchronous command

MTESt:SEGMent:POINt:REMove <MaskTestName>, <MaskSegmIdx>, <MaskSegmPtIdx>

Removes the specified point from the mask segment.

Setting parameters:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

<MaskSegmPtIdx> Number of the point. Counting starts from 0.

Usage: Setting only
Asynchronous command

MTESt:SEGMent:POINt:COUNt? <MaskTestName>, <MaskSegmIdx>

Returns the number of defined points for the specified mask segment.

Query parameters:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Usage: Query only

MTESt:SEGMent:POINt:X <MaskTestName>, <MaskSegmIdx>, <MaskSegmPtIdx>,<X>
MTESt:SEGMent:POINt:X? <MaskTestName>, <MaskSegmIdx>, <MaskSegmPtIdx>

Defines the x-value of the mask segment point.

Parameters:
<X>
Range: -100E+24 to 100E+24
Increment: 1E-6
*RST: 0
Default unit: s

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

<MaskSegmPtIdx> Number of the point. Counting starts from 0.

Example: See Chapter 24.3.3.1, "Creating a user mask", on page 1265

Usage: Asynchronous command

MTESt:SEGMent:POINt:Y <MaskTestName>, <MaskSegmIdx>, <MaskSegmPtIdx>,<Y>
MTESt:SEGMent:POINt:Y? <MaskTestName>, <MaskSegmIdx>, <MaskSegmPtIdx>

Defines the y-value of the mask segment point.

Parameters:
<Y>
Range: -100E+24 to 100E+24
Increment: 1E-6
*RST: 0
Default unit: V

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

<MaskSegmPtIdx> Number of the point. Counting starts from 0.

Example: See Chapter 24.3.3.1, "Creating a user mask", on page 1265

Usage: Asynchronous command

MTESt:SEGMent:RESCale:RECalculate <MaskTestName>, <MaskSegmIdx>

Multiplies and adds the given x- and y-factors and offsets to the coordinates of all points of the selected mask segment.

Setting parameters:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Usage: Setting only
Asynchronous command

MTESt:SEGMent:RESCale:XFACtor <MaskTestName>, <MaskSegmIdx>,<ExpansFactX>
MTESt:SEGMent:RESCale:XFACtor? <MaskTestName>, <MaskSegmIdx>
MTESt:SEGMent:RESCale:YFACtor <MaskTestName>, <MaskSegmIdx>,<ExpansFactY>
MTESt:SEGMent:RESCale:YFACtor? <MaskTestName>, <MaskSegmIdx>

Stretches or compresses the selected mask segment in horizontal (XFACtor) or vertical direction (YFACtor). The x- or y-values of all points of the selected mask segment are multiplied with this factor. Factors >1 stretch the mask segment, while factors between 0 and 1 compress it. Negative values are possible and change the algebraic sign.

Only takes effect after the `MTESt:SEGMent:RESCale:RECalculate` command.

Parameters:

<ExpansFactY> Range: -100 to 100
Increment: 1
*RST: 1

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Usage: Asynchronous command

MTESt:SEGMent:RESCale:XOFFset <MaskTestName>, <MaskSegmIdx>,<OffsetX>
MTESt:SEGMent:RESCale:XOFFset? <MaskTestName>, <MaskSegmIdx>

Moves the mask segment horizontally. The specified offset is added to the x-values of all points of the selected mask segment.

Only takes effect after the `MTESt:SEGMent:RESCale:RECalculate` command.

Parameters:

<OffsetX> Range: -50 to 50
Increment: 1E-9
*RST: 0
Default unit: s

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Usage: Asynchronous command

MTESt:SEGMent:RESCale:YOFFset <MaskTestName>, <MaskSegmIdx>,<OffsetY>
MTESt:SEGMent:RESCale:YOFFset? <MaskTestName>, <MaskSegmIdx>

Moves the mask segment vertically. The specified offset is added to the y-values of all points of the selected mask segment.

Only takes effect after the `MTESt:SEGMent:RESCale:RECalculate` command.

Parameters:

<OffsetY>
Range: -1000 to 1000
Increment: 1E-6
*RST: 0
Default unit: V

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

<MaskSegmIdx> Number of the segment. Counting starts from 0.

Usage: Asynchronous command

24.14.3 Mask definition: waveform mask

The chapter contains commands required for the definition of waveform masks - `MTESt:CTYPe` is set to `WFML`.

MTESt:REFWfm 1592
MTESt:WFMLupdate 1593
MTESt:WFMRescale:XWIDth 1593
MTESt:WFMRescale:YWIDth 1593
MTESt:WFMRescale:YPOSition 1594
MTESt:WFMRescale:YSTRetch 1594

MTESt:REFWfm <MaskTestName>,<Source>
MTESt:REFWfm? <MaskTestName>

Sets the reference waveform from which the mask is created.

The reference waveform can be created before, or loaded from a file with REFCurve commands, see Chapter 24.10.2, "Reference waveforms", on page 1446.

Parameters:

<Source>
REF1 | REFerence1 | REF2 | REFerence2 | REF3 | REFerence3 | REF4 | REFerence4

REF1 = REFerence1, REF2 = REFerence2, REF3 = REFerence3, REF4 = REFerence4: reference waveforms

*RST: REF1

Parameters for setting and query:

<MaskTestName> String containing the name of the mask test

Usage: Asynchronous command

MTESt:WFMLupdate <MaskTestName>

Creates the upper and lower mask limit from the envelope of the selected reference waveform. If the reference waveform was not defined before, it is created automatically from the mask test source waveform which is set wtih `MTESt:SOURce`.

Setting parameters:
<MaskTestName> String containing the name of the mask test

Usage: Setting only
Asynchronous command

MTESt:WFMRescale:XWIDth <MaskTestName>,<HorizontalWidth>
MTESt:WFMRescale:XWIDth? <MaskTestName>

Sets the width of the mask in horizontal direction. The specified factor in divisions is added to the positive x-values and subtracted from the negative x-values of the mask limits in relation to the source waveform of the mask.

Parameters:
<HorizontalWidth> Range: 0 to 1000
Increment: 0.01
*RST: 0
Default unit: div

Parameters for setting and query:
<MaskTestName> String containing the name of the mask test

Usage: Asynchronous command

MTESt:WFMRescale:YWIDth <MaskTestName>,<VerticalWidth>
MTESt:WFMRescale:YWIDth? <MaskTestName>

Sets the width of the waveform mask in vertical direction. The specified factor in divisions is added to the y-values of the upper mask limit and subtracted from the y-values of the lower mask limit. Thus, the upper half of the mask is pulled upwards, the lower half is pulled down.

Parameters:
<VerticalWidth> Vertical mask width in divisions
Range: 0 to 1000
Increment: 0.01
*RST: 0
Default unit: div

Parameters for setting and query:
<MaskTestName> String containing the name of the mask test

Usage: Asynchronous command

MTESt:WFMRescale:YPOSition <MaskTestName>,<VertPosi>
MTESt:WFMRescale:YPOSition? <MaskTestName>

Moves the mask vertically within the display.

Parameters:

<VertPosi>

Range: -1000 to 1000
Increment: 0.01
*RST: 0
Default unit: div

Parameters for setting and query:

<MaskTestName> String containing the name of the mask test

Usage: Asynchronous command

MTESt:WFMRescale:YSTRetch <MaskTestName>,<VerticalStretch>
MTESt:WFMRescale:YSTRetch? <MaskTestName>

Sets the vertical scaling to stretch the mask in y-direction. The scaling axis is the horizontal line through the lowest value of the lower mask limit.

Parameters:

<VerticalStretch> Scale factor in %

Range: 10 to 1000
Increment: 1
*RST: 100
Default unit: %

Parameters for setting and query:

<MaskTestName> String containing the name of the mask test

Usage: Asynchronous command

24.14.4 Event actions

MTESt:ONViolation:BEEP....1594
MTESt:ONViolation:STOP....1595
MTESt:ONViolation:PRINt....1595
MTESt:ONViolation:SAVewaveform....1595
MTESt:ONViolation:REPort....1596
MTESt:ONViolation:TRIGgerout....1596
MTESt:ONViolation:RUNexec....1596

MTESt:ONViolation:BEEP <MaskTestName>,<Beep>
MTESt:ONViolation:BEEP? <MaskTestName>

Generates a beep sound for the specified event.

Parameters:
<Beep> NOACtion | SUCCess | VIOLation
See Chapter 24.4.5, "Event parameter", on page 1286
*RST: NOACtion

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:ONViolation:STOP <MaskTestName>,<StopAcq>
MTESt:ONViolation:STOP? <MaskTestName>

Stops data acquisition for the specified event.

Parameters:
<StopAcq> NOACtion | SUCCess | VIOLation
See Chapter 24.4.5, "Event parameter", on page 1286
*RST: NOACtion

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:ONViolation:PRINt <MaskTestName>,<Print>
MTESt:ONViolation:PRINt? <MaskTestName>

Saves a screenshot including the measurement results for the specified event.

Parameters:
<Print> NOACtion | SUCCess | VIOLation
See Event parameter
*RST: NOACtion

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:ONViolation:SAVewaveform <MaskTestName>,<SaveWfm>
MTESt:ONViolation:SAVewaveform? <MaskTestName>

Saves the waveform data.

Parameters:
<SaveWfm> NOACtion | SUCCess | VIOLation
See Chapter 24.4.5, "Event parameter", on page 1286
*RST: NOACtion

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:ONViolation:REPort <MaskTestName>,<Report>
MTESt:ONViolation:REPort? <MaskTestName>

Creates and saves a report of the current settings and results.

Parameters:
<Report> NOACtion | SUCCess | VIOLation

See Chapter 24.4.5, "Event parameter", on page 1286

*RST: NOACtion

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:ONViolation:TRIGgerout <MaskTestName>,<TriggerOutPulse>
MTESt:ONViolation:TRIGgerout? <MaskTestName>

Creates a trigger out pulse on mask violation or successful completion of the test cycle.

Parameters:
<TriggerOutPulse> NOACtion | SUCCess | VIOLation

*RST: NOACtion

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:ONViolation:RUNexec <MaskTestName>,<RunExecutable>
MTESt:ONViolation:RUNexec? <MaskTestName>

Starts an external application if the command is set to `VIOLation` or `SUCCess`.

Use the following commands to set up the application:

- `EXECutable:NAME` on page 1668
- `EXECutable:PARameter` on page 1669
- `EXECutable:WDIRectory` on page 1669

Parameters:
<RunExecutable> NOACtion | SUCCess | VIOLation

See Chapter 24.4.5, "Event parameter", on page 1286

*RST: NOACtion

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

24.14.5 Results

MTESt:RESult:STATe? 1597
MTESt:RESult[:RESult]? 1597
MTESt:RESult:COUNt:WAVeforms? 1597
MTESt:RESult:COUNt:REMaining? 1598
MTESt:RESult:COUNt:FWAVeforms? 1598
MTESt:RESult:COUNt:FAILures? 1598
MTESt:RESult:FRATe? 1599

MTESt:RESult:STATe? <MaskTestName>

Shows if the test is running or has finished. The state is set to "Finished" if no acquisitions remain (see `MTESt:RESult:COUNt:REMaining?` on page 1598).

Query parameters:
<MaskTestName>

Return values:
<State> RUNNing | FINished
*RST: RUNNing

Usage: Query only
Asynchronous command

MTESt:RESult[:RESult]? <MaskTestName>

Returns the test result.

A test has failed if the number of sample hits or acquisition hits exceeds the limit of "Violation tolerance" hits (see `MTESt:TOLerance` on page 1584, `MTESt:RESult:COUNt:FAILures?` on page 1598 and `MTESt:RESult:COUNt:FWAVeforms?` on page 1598).

Query parameters:
<MaskTestName>

Return values:
<TestResult> PASS | FAIL
*RST: PASS

Usage: Query only
Asynchronous command

MTESt:RESult:COUNt:WAVeforms? <MaskTestName>

Returns the number of tested acquisitions.

Query parameters:
<MaskTestName>

Return values:
<AcqsCompleted> Range: 0 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

MTESt:RESult:COUNt:REMaining? <MaskTestName>

Remaining acquisitions until "Average count / Nx Single count" is reached.

The value is useful if you test a specified number of acquisitions with action "Stop acquisition" on violation, or if the acquisition has been stopped manually.

Query parameters:
<MaskTestName>

Return values:
<AcqsRemaining> Range: 0 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

MTESt:RESult:COUNt:FWAVeforms? <MaskTestName>

Returns the number of acquisitions that contained at least one sample hit.

Query parameters:
<MaskTestName>

Return values:
<AcquisitionHits> Range: 0 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

MTESt:RESult:COUNt:FAILures? <MaskTestName>

Returns the number of sample hits that violated the mask.

Query parameters:
<MaskTestName>

Return values:
<SampleHits> Range: 0 to 100E+24
*RST: 0

Usage: Query only
Asynchronous command

MTESt:RESult:FRATe? <MaskTestName>

Ratio of acquisition hits to the number of tested acquisitions.

Query parameters:
<MaskTestName>

Return values:

<FailRate>	Range:	-100E+24 to 100E+24
	*RST:	0
	Default unit:	%

Usage: Query only
Asynchronous command

24.14.6 Eye mask testing

24.14.6.1 Definition of eye masks

To perform mask testing on eye diagrams, the option R&S RTO6-K12 adds a special mask definition type to the common mask definitions: the definition type "Eye".

MTESt:CEMask 1599
MTESt:EYEMask:TYPE 1600
MTESt:EYEMask:HEIGht<m>[:VALue] 1600
MTESt:EYEMask:WIDTh<m>[:VALue] 1600
MTESt:EYEMask:MSKRight 1601
MTESt:EYEMask:MSKLeft 1601
MTESt:EYEMask:HPERiod 1601
MTESt:EYEMask:MSKTop 1601
MTESt:EYEMask:MSKBottom 1601
MTESt:EYEMask:TOFFset 1602
MTESt:EYEMask:BOFFset 1602
MTESt:EYEMask:TBSYmmetric 1602
MTESt:EYEMask:TBWidth 1602
MTESt:EYEMask:HPOSition 1603
MTESt:EYEMask:VPOSition 1603

MTESt:CEMask <MaskTestName>

Converts the test and mask definitions of the current mask test to a new mask test of type "User".

Setting parameters:

<MaskTestName>	String with the name of the mask test

Usage: Setting only
Asynchronous command

MTESt:EYEMask:TYPE <MaskTestName>,<Type>
MTESt:EYEMask:TYPE? <MaskTestName>

Defines the outline of the eye mask.

Parameters:
<Type> SQUare | DIAMond | HEXagon | OCTagon
*RST: DIAMond

Parameters for setting and query:
<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:EYEMask:HEIGht<m>[:VALue] <MaskTestName>, <value>

Defines the main heigth af all eye mask shapes and the minor height for octagon mask shapes.

Suffix:
<m> 1..2
1 - main width
2 - minor width

Parameters:
<value> Default unit: s

Setting parameters:
<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:EYEMask:WIDTh<m>[:VALue] <MaskTestName>, <value>

Defines the main width af all eye mask shapes and the minor width for hexagon and octagon mask shapes.

Suffix:
<m> 1..2
1 - main width
2 - minor width

Parameters:
<value> Default unit: s

Setting parameters:
<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:EYEMask:MSKRight <MaskTestName>,<Copy>
MTESt:EYEMask:MSKRight? <MaskTestName>
MTESt:EYEMask:MSKLeft <MaskTestName>,<Copy>
MTESt:EYEMask:MSKLeft? <MaskTestName>

Copies the eye shape to the right and left, respectively.

The distance of the copy is defined using `MTESt:EYEMask:HPERiod`.

Parameters:

<Copy> ON | OFF
*RST: OFF

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:EYEMask:HPERiod <MaskTestName>,<InterpattLength>
MTESt:EYEMask:HPERiod? <MaskTestName>

Defines the time distance between the shape centers if `MTESt:EYEMask:MSKLeft` and/or `MTESt:EYEMask:MSKRight` are ON.

Parameters:

<InterpattLength> Range: 0 to 100
Increment: 0.01
*RST: 0.5
Default unit: s

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:EYEMask:MSKTop <MaskTestName>,<Add>
MTESt:EYEMask:MSKTop? <MaskTestName>
MTESt:EYEMask:MSKBottom <MaskTestName>,<Add>
MTESt:EYEMask:MSKBottom? <MaskTestName>

Enable the upper (top) and lower (bottom) mask region, respectively.

Parameters:

<Add> ON | OFF
*RST: OFF

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:EYEMask:TOFFset <MaskTestName>,<RectDistTop>
MTESt:EYEMask:TOFFset? <MaskTestName>
MTESt:EYEMask:BOFFset <MaskTestName>,<RectDistBtom>
MTESt:EYEMask:BOFFset? <MaskTestName>

Voltage distance from the eye shape center that limit the upper (TOFFset) and lower (BOFFset) regions.

Parameters:

<RectDistBtom>	Range:	0 to 100
	Increment:	0.01
	*RST:	0.5
	Default unit:	V

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:EYEMask:TBSYmmetric <MaskTestName>,<TopBtomSymetrie>
MTESt:EYEMask:TBSYmmetric? <MaskTestName>

Sets bottom and top offsets to the same value so that the outer regions are symmetric to the eye shape.

Parameters:

<TopBtomSymetrie>	ON | OFF	
	*RST:	ON

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:EYEMask:TBWidth <MaskTestName>,<RectangleLength>
MTESt:EYEMask:TBWidth? <MaskTestName>

Sets the time width of the outer regions, symmetric to the eye shape center.

Parameters:

<RectangleLength>	Range:	0 to 100
	Increment:	0.01
	*RST:	0.5
	Default unit:	s

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:EYEMask:HPOSition <MaskTestName>,<PatOffsetX>
MTESt:EYEMask:HPOSition? <MaskTestName>

Sets the horizontal (time) value of the eye shape enter and thus defines the horizontal position of the eye shape on the display.

Parameters:

<PatOffsetX>	Range:	-100 to 100
	Increment:	0.01
	*RST:	0
	Default unit:	s

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

MTESt:EYEMask:VPOSition <MaskTestName>,<PatOffsetY>
MTESt:EYEMask:VPOSition? <MaskTestName>

Sets the vertical (voltage) value of the eye shape enter and thus defines the vertical position of the eye shape on the display.

Parameters:

<PatOffsetY>	Range:	-100 to 100
	Increment:	0.01
	*RST:	0
	Default unit:	V

Parameters for setting and query:

<MaskTestName> String with the name of the mask test

Usage: Asynchronous command

24.15 Search

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

- General search settings....1604
- Basic trigger search conditions....1606
- Edge search conditions....1608
- Glitch search conditions....1608
- Interval search conditions....1610
- Runt search conditions....1611
- Slew rate search conditions....1614
- Timeout search conditions....1616
- Width search conditions....1617
- Window search conditions....1619
- Data2clock search conditions....1621
- Pattern search conditions....1623
- State search conditions....1628
- Search on spectrum....1630
- Search gate settings....1631
- Search results....1634

24.15.1 General search settings

SEARch:ADD....1604
SEARch:CLEar....1604
SEARch:REMove....1605
SEARch:SOURce....1605
SEARch:ONLine....1605
SEARch:ALL....1605

SEARch:ADD <SearchName>

Creates a new search definition with the specified name.

Setting parameters:

<SearchName> String with the name of the search

Example: See Chapter 24.3.4.1, "Searching for a pulse of specified width", on page 1266

Usage: Setting only
Asynchronous command

SEARch:CLEar <SearchName>

Clears the search results once to start a new search.

Setting parameters:

<SearchName> Search definition

Usage: Setting only
Asynchronous command

SEARch:REMove <Key>

Deletes the specified search definition.

Setting parameters:

<Key> String with the name of the search

Usage: Setting only
Asynchronous command

SEARch:SOURce <SearchName>,<Source>
SEARch:SOURce? <SearchName>

Defines the source on which the search conditions are applied. The source can be any analog or digital channel, math or reference waveform as well as a serial bus configured for a supported protocol.

Parameters:

<Source> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | SBUS1 | SBUS2 | SBUS3 | SBUS4

Source of the search, see Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C1W1

Parameters for setting and query:

<SearchName> String with the name of the search

Example: See Chapter 24.3.4.1, "Searching for a pulse of specified width", on page 1266

Usage: Asynchronous command

SEARch:ONLine <SearchName>,<OnlineState>
SEARch:ONLine? <SearchName>

If enabled, a search is performed repeatedly for each new data acquisition.

Parameters:

<OnlineState> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:ALL <SearchName>

Performs a search for all results on the existing data from the selected source.

Setting parameters:
<SearchName> Search definition

Example: See Chapter 24.3.4.1, "Searching for a pulse of specified width", on page 1266

Usage: Setting only
Asynchronous command

24.15.2 Basic trigger search conditions

SEARch:TRIGger:EDGE[:STATe]....1606
SEARch:TRIGger:GLITch[:STATe]....1606
SEARch:TRIGger:INTerval[:STATe]....1606
SEARch:TRIGger:PATTern[:STATe]....1606
SEARch:TRIGger:RUNT[:STATe]....1606
SEARch:TRIGger:SETHold[:STATe]....1606
SEARch:TRIGger:SLEWrate[:STATe]....1607
SEARch:TRIGger:STATe[:STATe]....1607
SEARch:TRIGger:TIMeout[:STATe]....1607
SEARch:TRIGger:WIDTh[:STATe]....1607
SEARch:TRIGger:WINDow[:STATe]....1607
SEARch:TRIGger:LEVel[:VALue]....1607
SEARch:TRIGger:EDGE:ACOPy....1607
SEARch:TRIGger:GLITch:ACOPy....1607
SEARch:TRIGger:INTerval:ACOPy....1607
SEARch:TRIGger:PATTern:ACOPy....1607
SEARch:TRIGger:RUNT:ACOPy....1607
SEARch:TRIGger:SETHold:ACOPy....1607
SEARch:TRIGger:SLEWrate:ACOPy....1607
SEARch:TRIGger:STATe:ACOPy....1608
SEARch:TRIGger:TIMeout:ACOPy....1608
SEARch:TRIGger:WIDTh:ACOPy....1608
SEARch:TRIGger:WINDow:ACOPy....1608
SEARch:TRIGger:EDGE:BCOPy....1608

SEARch:TRIGger:EDGE[:STATe] <SearchName>,<State>
SEARch:TRIGger:EDGE[:STATe]? <SearchName>
SEARch:TRIGger:GLITch[:STATe] <SearchName>,<State>
SEARch:TRIGger:GLITch[:STATe]? <SearchName>
SEARch:TRIGger:INTerval[:STATe] <SearchName>,<State>
SEARch:TRIGger:INTerval[:STATe]? <SearchName>
SEARch:TRIGger:PATTern[:STATe] <SearchName>,<State>
SEARch:TRIGger:PATTern[:STATe]? <SearchName>
SEARch:TRIGger:RUNT[:STATe] <SearchName>,<State>
SEARch:TRIGger:RUNT[:STATe]? <SearchName>
SEARch:TRIGger:SETHold[:STATe] <SearchName>,<State>
SEARch:TRIGger:SETHold[:STATe]? <SearchName>

SEARch:TRIGger:SLEWrate[:STATe] <SearchName>,<State>
SEARch:TRIGger:SLEWrate[:STATe]? <SearchName>
SEARch:TRIGger:STATe[:STATe] <SearchName>,<State>
SEARch:TRIGger:STATe[:STATe]? <SearchName>
SEARch:TRIGger:TIMeout[:STATe] <SearchName>,<State>
SEARch:TRIGger:TIMeout[:STATe]? <SearchName>
SEARch:TRIGger:WIDTh[:STATe] <SearchName>,<State>
SEARch:TRIGger:WIDTh[:STATe]? <SearchName>
SEARch:TRIGger:WINDow[:STATe] <SearchName>,<State>
SEARch:TRIGger:WINDow[:STATe]? <SearchName>

Includes the search conditions for the selected trigger event type in the next search.

Parameters:

<State> ON | OFF

*RST: OFF

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:LEVel[:VALue] <SearchName>, <SignalSource>, <Value>
SEARch:TRIGger:LEVel[:VALue]? <Key>, <SignalSource>

Sets the voltage of the trigger level that is used to determine other parameters.

Parameters:

<Value> Voltage value

Parameters for setting and query:

<SearchName> String with the name of the search

<SignalSource> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4

Source of the search, see Chapter 24.4.2, "Waveform parameter", on page 1284

Usage: Asynchronous command

SEARch:TRIGger:EDGE:ACOPy <SearchName>
SEARch:TRIGger:GLITch:ACOPy <SearchName>
SEARch:TRIGger:INTerval:ACOPy <SearchName>
SEARch:TRIGger:PATTern:ACOPy <SearchName>
SEARch:TRIGger:RUNT:ACOPy <SearchName>
SEARch:TRIGger:SETHold:ACOPy <SearchName>
SEARch:TRIGger:SLEWrate:ACOPy <SearchName>

SEARch:TRIGger:STATe:ACOPy <SearchName>
SEARch:TRIGger:TIMeout:ACOPy <SearchName>
SEARch:TRIGger:WIDTh:ACOPy <SearchName>
SEARch:TRIGger:WINDow:ACOPy <SearchName>

Copies the trigger event configuration from Trigger A for the selected channel source to the search condition settings.

See Chapter 6.3, "Trigger types", on page 193.

Setting parameters:
<SearchName> Search definition

Usage: Setting only
Asynchronous command

SEARch:TRIGger:EDGE:BCOPy <SearchName>

Copies the trigger event configuration from trigger B for the selected channel source to the search condition settings.

Setting parameters:
<SearchName> String with name of the search

Usage: Setting only
Asynchronous command

24.15.3 Edge search conditions

Trigger level setting: `SEARch:TRIGger:LEVel[:VALue]` on page 1607

SEARch:TRIGger:EDGE:SLOPe..... 1608

SEARch:TRIGger:EDGE:SLOPe <SearchName>,<Slope>
SEARch:TRIGger:EDGE:SLOPe? <SearchName>

Selects the edge type.

Parameters:
<Slope> POSitive | NEGative | EITHer

See Chapter 24.4.3, "Slope parameter", on page 1285.

*RST: POSitive

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

24.15.4 Glitch search conditions

Trigger level setting: `SEARch:TRIGger:LEVel[:VALue]` on page 1607

SEARch:TRIGger:GLITch:POLarity.....1609
SEARch:TRIGger:GLITch:RANGe.....1609
SEARch:TRIGger:GLITch:WIDTh.....1609

SEARch:TRIGger:GLITch:POLarity <SearchName>,<Polarity>
SEARch:TRIGger:GLITch:POLarity? <SearchName>

Indicates the polarity of a pulse, that is the direction of the first pulse slope.

Parameters:

<Polarity> POSitive | NEGative | EITHer

See Chapter 24.4.4, "Polarity parameter", on page 1285.

*RST: POSitive

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:GLITch:RANGe <SearchName>,<RangeMode>
SEARch:TRIGger:GLITch:RANGe? <SearchName>

Selects which glitches are identified: shorter or longer than the specified width (see `SEARch:TRIGger:GLITch:WIDTh` on page 1609).

Parameters:

<RangeMode> SHORter | LONGer

*RST: SHORter

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:GLITch:WIDTh <SearchName>,<Width>
SEARch:TRIGger:GLITch:WIDTh? <SearchName>

Sets the length of a glitch. The instrument triggers on pulses shorter or longer than this value (see also `SEARch:TRIGger:GLITch:RANGe` on page 1609).

You need to know the expected pulse widths of the circuit to set the glitch width correctly.

Parameters:

<Width> Range: 100E-12 to 1E-3
Increment: 100E-6
*RST: 1E-9
Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

24.15.5 Interval search conditions

Trigger level setting: `SEARch:TRIGger:LEVel[:VALue]` on page 1607

SEARch:TRIGger:INTerval:SLOPe....1610
SEARch:TRIGger:INTerval:DELTa....1610
SEARch:TRIGger:INTerval:RANGe....1610
SEARch:TRIGger:INTerval:WIDTh....1611

SEARch:TRIGger:INTerval:SLOPe <SearchName>,<Slope>
SEARch:TRIGger:INTerval:SLOPe? <SearchName>

Sets the edge for the search.

Parameters:

<Slope> POSitive | NEGative | EITHer

See Chapter 24.4.3, "Slope parameter", on page 1285.

*RST: POSitive

Parameters for setting and query:

<SearchName> String parameter, name of the search definition

Usage: Asynchronous command

SEARch:TRIGger:INTerval:DELTa <SearchName>,<WidthDelta>
SEARch:TRIGger:INTerval:DELTa? <SearchName>

Defines a range around the "Interval width" value (see `SEARch:TRIGger:INTerval:WIDTh` on page 1611).

Parameters:

<WidthDelta> Range: 0 to 10
Increment: 100E-9
*RST: 0
Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:INTerval:RANGe <SearchName>,<RangeMode>
SEARch:TRIGger:INTerval:RANGe? <SearchName>

Selects how the range of an interval is defined based on the interval width and delta (see `SEARch:TRIGger:INTerval:WIDTh` on page 1611 and `SEARch:TRIGger:INTerval:DELTa` on page 1610).

Parameters:

<RangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers on pulse intervals inside a given range. The range is defined by "Interv. width" and "±Delta".

OUTSide
Triggers on intervals outside a given range. The range definition is the same as for "Within" range.

SHORter
Triggers on intervals shorter than the given "Interv. width".

LONGer
Triggers on intervals longer than the given "Interv. width".

*RST: OUTSide

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:INTerval:WIDTh <SearchName>,<Width>
SEARch:TRIGger:INTerval:WIDTh? <SearchName>

Defines the time between two pulses.

Parameters:
<Width>
Range: 100E-12 to 864
Increment: 100E-9
*RST: 5E-9
Default unit: s

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

24.15.6 Runt search conditions

SEARch:TRIGger:RUNT:DELTa 1611
SEARch:TRIGger:RUNT:POLarity 1612
SEARch:TRIGger:RUNT:RANGe 1612
SEARch:TRIGger:RUNT:WIDTh 1613
SEARch:TRIGger:LEVel:RUNT:LOWer 1613
SEARch:TRIGger:LEVel:RUNT:UPPer 1613

SEARch:TRIGger:RUNT:DELTa <SearchName>,<WidthDelta>
SEARch:TRIGger:RUNT:DELTa? <SearchName>

Defines a range around the given runt width.

Parameters:

<WidthDelta> Range: 100E-12 to 864
Increment: 100E-9
*RST: 100E-12
Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:RUNT:POLarity <SearchName>,<Polarity>
SEARch:TRIGger:RUNT:POLarity? <SearchName>

Indicates the polarity of a pulse, that is the direction of the first pulse slope.

Parameters:

<Polarity> POSitive | NEGative | EITHer

See Chapter 24.4.4, "Polarity parameter", on page 1285.

*RST: POSitive

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:RUNT:RANGe <SearchName>,<Mode>
SEARch:TRIGger:RUNT:RANGe? <SearchName>

Selects how the time limit of the runt pulse is defined based on the runt width and delta (see `SEARch:TRIGger:RUNT:WIDTh` on page 1613 and `SEARch:TRIGger:RUNT:DELTa` on page 1611).

Parameters:

<Mode> ANY | LONGer | SHORter | WITHin | OUTSide

ANY
Triggers on all runts fulfilling the level condition, without time limitation.

LONGer
Triggers on runts longer than the given "Runt width".

SHORter
Triggers on runts shorter than the given "Runt width".

WITHin
Triggers if the runt length is inside a given time range. The range is defined by "Runt width" and "±Delta".

OUTSide
Triggers if the runt length is outside a given time range. The range definition is the same as for "Within" range.

*RST: ANY

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:RUNT:WIDTh <SearchName>,<Width>
SEARch:TRIGger:RUNT:WIDTh? <SearchName>

For the ranges "Shorter" and "Longer", the runt width defines the maximum and minimum pulse width, respectively.

For the ranges "Within" and "Outside", the runt width defines the center of a range which is defined by "±Delta".

The range is defined using `SEARch:TRIGger:RUNT:RANGe` on page 1612.

Parameters:
<Width>
Range: 100E-12 to 864
Increment: 100E-9
*RST: 5E-9
Default unit: s

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:LEVel:RUNT:LOWer <SearchName>, <SignalSource>, <Value>
SEARch:TRIGger:LEVel:RUNT:LOWer? <Key>, <SignalSource>
SEARch:TRIGger:LEVel:RUNT:UPPer <SearchName>, <SignalSource>, <Value>
SEARch:TRIGger:LEVel:RUNT:UPPer? <Key>, <SignalSource>

Set the lower and upper voltage threshold, respectively.

Parameters:
<Value> Voltage value

Parameters for setting and query:
<SearchName> String with the name of the search

<SignalSource> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4

Source of the search, see Chapter 24.4.2, "Waveform parameter", on page 1284

Usage: Asynchronous command

24.15.7 Slew rate search conditions

SEARch:TRIGger:SLEWrate:DELTa....1614
SEARch:TRIGger:SLEWrate:RANGe....1614
SEARch:TRIGger:SLEWrate:SLOPe....1615
SEARch:TRIGger:SLEWrate:TIME....1615
SEARch:TRIGger:LEVel:TRANsition:LOWer....1615
SEARch:TRIGger:LEVel:TRANsition:UPPer....1615

SEARch:TRIGger:SLEWrate:DELTa <SearchName>,<TimeDelta>
SEARch:TRIGger:SLEWrate:DELTa? <SearchName>

Defines a time range around the given slew rate.

Parameters:

<TimeDelta>
Range: 0 to 10
Increment: 100E-9
*RST: 0
Default unit: s

Parameters for setting and query:

<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:SLEWrate:RANGe <SearchName>,<RangeMode>
SEARch:TRIGger:SLEWrate:RANGe? <SearchName>

Selects how the time limit for the slew rate is defined. The time measurement starts when the signal crosses the first trigger level - the upper or lower level depending on the selected slope - and stops when the signal crosses the second level.

Parameters:

<RangeMode> INSRange | OUTRange | LTHan | GTHan

INSRange
Triggers on slew rates inside a given time range. The range is defined by "Slew rate" and "±Delta".

OUTRange
Triggers on slew rates outside a given time range. The range definition is the same as for "Within" range.

LTHan
Triggers on slew rates shorter than the given "Slew rate" limit.

GTHan
Triggers on slew rates longer than the given "Slew rate" limit.

*RST: GTHan

Parameters for setting and query:

<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:SLEWrate:SLOPe <SearchName>,<Slope>
SEARch:TRIGger:SLEWrate:SLOPe? <SearchName>

Selects the edge type.

Parameters:

<Slope> POSitive | NEGative | EITHer

See Chapter 24.4.3, "Slope parameter", on page 1285.

*RST: POSitive

Parameters for setting and query:

<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:SLEWrate:TIME <SearchName>,<Time>
SEARch:TRIGger:SLEWrate:TIME? <SearchName>

For the ranges "Within" and "Outside", the slew rate defines the center of a range which is defined by the limits "±Delta".

For the ranges "Shorter" and "Longer", the slew rate defines the maximum and minimum slew rate limits, respectively.

The range is defined using `SEARch:TRIGger:SLEWrate:RANGe`.

Parameters:

<Time> Range: 100E-12 to 864
Increment: 100E-9
*RST: 100E-12
Default unit: s

Parameters for setting and query:

<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:LEVel:TRANsition:LOWer <SearchName>, <SignalSource>, <Value>
SEARch:TRIGger:LEVel:TRANsition:LOWer? <Key>, <SignalSource>
SEARch:TRIGger:LEVel:TRANsition:UPPer <SearchName>, <SignalSource>, <Value>
SEARch:TRIGger:LEVel:TRANsition:UPPer? <Key>, <SignalSource>

Set the lower and upper voltage thresholds, respectively. When the signal crosses this level, the slew rate measurement starts or stops depending on the selected slope.

Parameters:

<Value> Voltage value

Parameters for setting and query:

<SearchName> String with the name of the search

<SignalSource> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4

Source of the search, see Chapter 24.4.2, "Waveform parameter", on page 1284

Usage: Asynchronous command

24.15.8 Timeout search conditions

Trigger level setting: `SEARch:TRIGger:LEVel[:VALue]` on page 1607

SEARch:TRIGger:TIMeout:RANGe....1616
SEARch:TRIGger:TIMeout:TIME....1616

SEARch:TRIGger:TIMeout:RANGe <SearchName>,<TimeoutMode>
SEARch:TRIGger:TIMeout:RANGe? <SearchName>

Selects the relation of the signal level to the trigger level:

Parameters:

<TimeoutMode> HIGH | LOW | EITHer

HIGH
The signal level stays above the trigger level.

LOW
The signal level stays below the trigger level.

EITHer
The signal level stays above or below the trigger level.

*RST: HIGH

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:TIMeout:TIME <SearchName>,<Time>
SEARch:TRIGger:TIMeout:TIME? <SearchName>

Defines the time limit for the timeout at which the instrument triggers.

Parameters:

<Time>
Range: 100E-12 to 864
Increment: 100E-9
*RST: 100E-9
Default unit: s

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

24.15.9 Width search conditions

Trigger level setting: `SEARch:TRIGger:LEVel[:VALue]` on page 1607

SEARch:TRIGger:WIDTh:DELTa..... 1617
SEARch:TRIGger:WIDTh:POLarity..... 1617
SEARch:TRIGger:WIDTh:RANGe.....1618
SEARch:TRIGger:WIDTh:WIDTh.....1618

SEARch:TRIGger:WIDTh:DELTa <SearchName>,<WidthDelta>
SEARch:TRIGger:WIDTh:DELTa? <SearchName>

Defines a range around the given width value (see also `SEARch:TRIGger:WIDTh:WIDTh` on page 1618).

Parameters:
<WidthDelta> Range: 0 to 432
Increment: 500E-12
*RST: 0
Default unit: s

Parameters for setting and query:
<SearchName> Search definition

Example: See Chapter 24.3.4.1, "Searching for a pulse of specified width", on page 1266

Usage: Asynchronous command

SEARch:TRIGger:WIDTh:POLarity <SearchName>,<Polarity>
SEARch:TRIGger:WIDTh:POLarity? <SearchName>

Indicates the polarity of a pulse, that is the direction of the first pulse slope.

Parameters:
<Polarity> POSitive | NEGative | EITHer
See Chapter 24.4.4, "Polarity parameter", on page 1285.
*RST: POSitive

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:WIDTh:RANGe <SearchName>,<RangeMode>
SEARch:TRIGger:WIDTh:RANGe? <SearchName>

Selects how the range of a pulse width is defined in relation to the width and delta (see `SEARch:TRIGger:WIDTh:WIDTh` on page 1618 and `SEARch:TRIGger:WIDTh:DELTa` on page 1617).

Parameters:

<RangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers on pulses inside a given range. The range of the pulse width is defined by "Width" and "±Delta".

OUTSide
Triggers on pulses outside a given range. The range definition is the same as for "Within" range.

SHORter
Triggers on pulses shorter than the given "Width".

LONGer
Triggers on pulses longer than the given "Width".

*RST: WITHin

Parameters for setting and query:

<SearchName> Search definition

Example: See Chapter 24.3.4.1, "Searching for a pulse of specified width", on page 1266

Usage: Asynchronous command

SEARch:TRIGger:WIDTh:WIDTh <SearchName>,<Width>
SEARch:TRIGger:WIDTh:WIDTh? <SearchName>

For the ranges "Within" and "Outside", the width defines the center of a range which is defined by the limits "±Delta".

For the ranges "Shorter" and "Longer", the width defines the maximum and minimum pulse width, respectively.

The range is defined using `SEARch:TRIGger:WIDTh:RANGe`.

Parameters:

<Width> Range: 100E-12 to 864
Increment: 100E-9
*RST: 5E-9
Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Example: See Chapter 24.3.4.1, "Searching for a pulse of specified width", on page 1266

Usage: Asynchronous command

24.15.10 Window search conditions

SEARch:TRIGger:WINDow:DELTa....1619
SEARch:TRIGger:WINDow:RANGe....1619
SEARch:TRIGger:WINDow:TIMerange....1620
SEARch:TRIGger:WINDow:WIDTh....1620
SEARch:TRIGger:LEVel:WINDow:LOWer....1621
SEARch:TRIGger:LEVel:WINDow:UPPer....1621

SEARch:TRIGger:WINDow:DELTa <SearchName>,<WidthDelta>
SEARch:TRIGger:WINDow:DELTa? <SearchName>

Defines a range around the "Width" value (see `SEARch:TRIGger:WINDow:WIDTh` on page 1620).

Parameters:

<WidthDelta>
Range: 0 to 432
Increment: 500E-12
*RST: 0
Default unit: s

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:WINDow:RANGe <SearchName>,<RangeMode>
SEARch:TRIGger:WINDow:RANGe? <SearchName>

Selects how the signal run is compared with the window.

Parameters:

<RangeMode> ENTer | EXIT | WITHin | OUTSide

ENTer
Triggers when the signal crosses the upper or lower level and thus enters the window made up of these two levels.

EXIT
Triggers when the signal leaves the window.

WITHin
Triggers if the signal stays between the upper and lower level for a specified time. The time is defined in various ways by the `SEARch:TRIGger:WINDow:TIMerange` command.

OUTSide
Triggers if the signal stays above the upper level or below the lower level for a specified time. The time is also defined by the `SEARch:TRIGger:WINDow:TIMerange` command.

*RST: ENTer

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:WINDow:TIMerange <SearchName>,<TimeRangeMode>
SEARch:TRIGger:WINDow:TIMerange? <SearchName>

Selects how the time limit of the window is defined. Time conditioning is available for the vertical conditions "WITHin" and "OUTSide" (see `SEARch:TRIGger:WINDow:RANGe` on page 1619).

Parameters:

<TimeRangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers if the signal stays inside or outside the vertical window limits at least for the time *Width - Delta* and for *Width + Delta* at the most.

OUTSide
"Outside" is the opposite definition of "Within". The instrument triggers if the signal stays inside or outside the vertical window limits for a time shorter than *Width - Delta* or longer than *Width + Delta*.

SHORter
Triggers if the signal crosses vertical limits before the specified "Width" time is reached.

LONGer
Triggers if the signal crosses vertical limits before the specified "Width" time is reached.

*RST: WITHin

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:WINDow:WIDTh <SearchName>,<Width>
SEARch:TRIGger:WINDow:WIDTh? <SearchName>

For the ranges "Within" and "Outside", the width defines the center of a time range which is defined by the limits "±Delta".

For the ranges "Shorter" and "Longer", it defines the maximum and minimum time lapse, respectively.

The range is defined using `SEARch:TRIGger:WINDow:RANGe`.

Parameters:

<Width>
Range: 100E-12 to 864
Increment: 100E-9
*RST: 5E-9
Default unit: s

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:LEVel:WINDow:LOWer <SearchName>, <SignalSource>, <Value>
SEARch:TRIGger:LEVel:WINDow:LOWer? <Key>, <SignalSource>
SEARch:TRIGger:LEVel:WINDow:UPPer <SearchName>, <SignalSource>, <Value>
SEARch:TRIGger:LEVel:WINDow:UPPer? <Key>, <SignalSource>

Set the lower and upper voltage limits for the window.

Parameters:
<Value> Voltage value

Parameters for setting and query:
<SearchName> String with the name of the search

<SignalSource> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4

Source of the search, see Chapter 24.4.2, "Waveform parameter", on page 1284

Usage: Asynchronous command

24.15.11 Data2clock search conditions

Data level setting: `SEARch:TRIGger:LEVel[:VALue]` on page 1607

SEARch:TRIGger:SETHold:CEDGe....1621
SEARch:TRIGger:SETHold:CLEVel....1622
SEARch:TRIGger:SETHold:CSOurce....1622
SEARch:TRIGger:SETHold:HTIMe....1622
SEARch:TRIGger:SETHold:STIMe....1623

SEARch:TRIGger:SETHold:CEDGe <SearchName>,<ClockEdge>
SEARch:TRIGger:SETHold:CEDGe? <SearchName>

Sets the edge of the clock signal to define the time reference point for the setup and hold time.

Parameters:
<ClockEdge> POSitive | NEGative | EITHer

See Chapter 24.4.3, "Slope parameter", on page 1285.

*RST: POSitive

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:SETHold:CLEVel <SearchName>,<ClockLevel>
SEARch:TRIGger:SETHold:CLEVel? <SearchName>

Sets the voltage level for the clock signal. Both this command and `SEARch:TRIGger:SETHold:CEDGe` define the starting point for calculation of the setup and hold time.

Parameters:
<ClockLevel> Range: -10 to 10
Increment: 1E-3
*RST: 0
Default unit: V

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:SETHold:CSOurce <SearchName>,<ClockSource>
SEARch:TRIGger:SETHold:CSOurce? <SearchName>

Selects the waveform used for the clock signal.

Parameters:
<ClockSource> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

Source of the clock signal, see Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C1W1

Parameters for setting and query:
<SearchName> Search definition name

Usage: Asynchronous command

SEARch:TRIGger:SETHold:HTIMe <SearchName>,<HoldTime>
SEARch:TRIGger:SETHold:HTIMe? <SearchName>

Sets the minimum time **after** the clock edge while the data signal must stay steady above or below the data level.

The hold time can be negative. In this case, the setup time is always positive. The setup/hold interval starts before the clock edge (setup time) and ends before the clock edge (hold time). If you change the negative hold time, the setup time is adjusted by the intrument.

Parameters:

<HoldTime>	Range:	-99.999E-9 to 0.1
	Increment:	1E-9
	*RST:	0
	Default unit:	s

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:TRIGger:SETHold:STIMe <SearchName>,<SetupTime>
SEARch:TRIGger:SETHold:STIMe? <SearchName>

Sets the minimum time **before** the clock edge while the data signal must stay steady above or below the data level.

The setup time can be negative. In this case, the hold time is always positive. The setup/hold interval starts after the clock edge (setup time) and ends after the clock edge (hold time). If you change the negative setup time, the hold time is adjusted by the intrument.

Parameters:

<SetupTime>	Range:	-99.999E-9 to 0.1
	Increment:	1E-9
	*RST:	0
	Default unit:	s

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

24.15.12 Pattern search conditions

SEARch:TRIGger:PATTern:A[:ENABle]....1624
SEARch:TRIGger:PATTern:B[:ENABle]....1624
SEARch:TRIGger:PATTern:C[:ENABle]....1624
SEARch:TRIGger:PATTern:D[:ENABle]....1624
SEARch:TRIGger:PATTern:A:LOGic....1624
SEARch:TRIGger:PATTern:B:LOGic....1624
SEARch:TRIGger:PATTern:C:LOGic....1624
SEARch:TRIGger:PATTern:D:LOGic....1624
SEARch:TRIGger:PATTern:AB:LOGic....1625
SEARch:TRIGger:PATTern:CD:LOGic....1625
SEARch:TRIGger:PATTern:ABCD:LOGic....1625
SEARch:TRIGger:PATTern:MODE....1625
SEARch:TRIGger:PATTern:TIMeout:MODE....1626
SEARch:TRIGger:PATTern:TIMeout[:TIME]....1626

SEARch:TRIGger:PATTern:WIDTh:RANGe....1626
SEARch:TRIGger:PATTern:WIDTh[:WIDTh]....1627
SEARch:TRIGger:PATTern:WIDTh:DELTa....1627

SEARch:TRIGger:PATTern:A[:ENABle] <Searchname>, <State>
SEARch:TRIGger:PATTern:A[:ENABle]? <Searchname>
SEARch:TRIGger:PATTern:B[:ENABle] <Searchname>, <State>
SEARch:TRIGger:PATTern:B[:ENABle]? <Searchname>
SEARch:TRIGger:PATTern:C[:ENABle] <Searchname>, <State>
SEARch:TRIGger:PATTern:C[:ENABle]? <Searchname>
SEARch:TRIGger:PATTern:D[:ENABle] <Searchname>, <State>
SEARch:TRIGger:PATTern:D[:ENABle]? <Searchname>

Enables the channel to be considered in the pattern search. The trigger source channel is selected by default.

- A[:ENABle]: CH1
- B[:ENABle]: CH2
- C[:ENABle]: CH3
- D[:ENABle]: CH4

Digital channels are not available.

Parameters:
<State> ON | OFF

Parameters for setting and query:
<Searchname> String with name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:A:LOGic <Searchname>, <Operator>
SEARch:TRIGger:PATTern:A:LOGic? <Searchname>
SEARch:TRIGger:PATTern:B:LOGic <Searchname>, <Operator>
SEARch:TRIGger:PATTern:B:LOGic? <Searchname>
SEARch:TRIGger:PATTern:C:LOGic <Searchname>, <Operator>
SEARch:TRIGger:PATTern:C:LOGic? <Searchname>
SEARch:TRIGger:PATTern:D:LOGic <Searchname>, <Operator>
SEARch:TRIGger:PATTern:D:LOGic? <Searchname>

Defines the logic for the indicated channel:

- A: CH1
- B: CH2
- C: CH3
- D: CH4

Parameters:
<Operator> DIRect | HIGH | NOT | LOW

DIRect
Input value remains unchanged

NOT
Input value is inverted

Parameters for setting and query:
<Searchname> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:AB:LOGic <Searchname>, <Operator>
SEARch:TRIGger:PATTern:AB:LOGic? <Searchname>
SEARch:TRIGger:PATTern:CD:LOGic <Searchname>, <Operator>
SEARch:TRIGger:PATTern:CD:LOGic? <Searchname>
SEARch:TRIGger:PATTern:ABCD:LOGic <Searchname>, <Operator>
SEARch:TRIGger:PATTern:ABCD:LOGic? <Searchname>

Defines the logical combination of the indicated channels after evaluating the previous logical operations:

- AB: CH1 and CH2
- CD: CH3 and CH4
- ABCD: result of AB and CD

Parameters:
<Operator> AND | NAND | OR | NOR

AND: logical AND, conjunctive combination
NAND:logical NOT AND
OR:logical OR, disjunctive combination
NOR: logical NOT OR

Parameters for setting and query:
<Searchname> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:MODE <SearchName>,<Mode>
SEARch:TRIGger:PATTern:MODE? <SearchName>

Adds additional time limitation to the pattern definition.

Parameters:
<Mode> OFF | TIMeout | WIDTh

OFF
No time limitation. The event is found if the pattern condition is fulfilled.

TIMeout
Defines how long the result of the pattern condition stays high or low. The duration of the timeout is defined using `SEARch:TRIGger:PATTern:TIMeout[:TIME]` The result state is defined using `SEARch:TRIGger:PATTern:TIMeout:MODE`.

WIDTh
Defines a time range for keeping up the true result of the pattern condition. The range is defined using `SEARch:TRIGger:PATTern:WIDTh:RANGe`.

*RST: OFF

Parameters for setting and query:
<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:TIMeout:MODE <SearchName>,<TimeoutMode>
SEARch:TRIGger:PATTern:TIMeout:MODE? <SearchName>

Defines the condition for the timeout.

Parameters:
<TimeoutMode> HIGH | LOW | EITHer

EITHer
High or low, the pattern remains unchanged for the given time.

*RST: HIGH

Parameters for setting and query:
<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:TIMeout[:TIME] <SearchName>,<Time>
SEARch:TRIGger:PATTern:TIMeout[:TIME]? <SearchName>

Defines how long the result of the pattern condition must keep the given state.

Parameters:
<Time> Range: 100E-12 to 864
Increment: 100E-9
*RST: 100E-9
Default unit: s

Parameters for setting and query:
<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:WIDTh:RANGe <SearchName>,<WidthRangeMode>
SEARch:TRIGger:PATTern:WIDTh:RANGe? <SearchName>

Defines the time range of a pulse width for keeping up the true result of the pattern condition. The width and delta are specified using `SEARch:TRIGger:PATTern:WIDTh[:WIDTh]` and `SEARch:TRIGger:PATTern:WIDTh:DELTa`.

Parameters:
<WidthRangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin | OUTSide
Triggers on pulses inside or outside a given range. The range is defined by the width ±delta.

SHORter | LONGer
Triggers on pulses shorter or longer than the given width.

*RST: WITHin

Parameters for setting and query:
<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:WIDTh[:WIDTh] <SearchName>,<Width>
SEARch:TRIGger:PATTern:WIDTh[:WIDTh]? <SearchName>

For the ranges WITHin and OUTSide, the width defines the center of a range that is defined by the limits ±delta.

For the ranges SHORter and LONGer, the width defines the maximum and minimum pulse width, respectively.

To set the range mode, use `SEARch:TRIGger:PATTern:WIDTh:RANGe`. To set the delta value, use `SEARch:TRIGger:PATTern:WIDTh:DELTa`.

Parameters:
<Width>
Range: 100E-12 to 864
Increment: 100E-9
*RST: 5E-9
Default unit: s

Parameters for setting and query:
<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:PATTern:WIDTh:DELTa <SearchName>,<WidthDelta>
SEARch:TRIGger:PATTern:WIDTh:DELTa? <SearchName>

Defines a range around the width value specified using `SEARch:TRIGger:PATTern:WIDTh[:WIDTh]`.

Parameters:
<WidthDelta>
Range: 0 to 432
Increment: 500E-12
*RST: 0
Default unit: s

Parameters for setting and query:
<SearchName> String with the name of the search

Usage: Asynchronous command

24.15.13 State search conditions

SEARch:TRIGger:STATe:CSOurce 1628
SEARch:TRIGger:STATe:CEDGe 1628
SEARch:TRIGger:STATe:CLEVel 1628
SEARch:TRIGger:STATe:A[:ENABle] 1629
SEARch:TRIGger:STATe:B[:ENABle] 1629
SEARch:TRIGger:STATe:C[:ENABle] 1629
SEARch:TRIGger:STATe:D[:ENABle] 1629
SEARch:TRIGger:STATe:A:LOGic 1629
SEARch:TRIGger:STATe:B:LOGic 1629
SEARch:TRIGger:STATe:C:LOGic 1629
SEARch:TRIGger:STATe:D:LOGic 1629
SEARch:TRIGger:STATe:AB:LOGic 1630
SEARch:TRIGger:STATe:CD:LOGic 1630
SEARch:TRIGger:STATe:ABCD:LOGic 1630

SEARch:TRIGger:STATe:CSOurce <SearchName>,<Source>
SEARch:TRIGger:STATe:CSOurce? <SearchName>

Sets the source of the clock signal.

Parameters:

<Source> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Parameters for setting and query:

<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:STATe:CEDGe <SearchName>,<ClockEdge>
SEARch:TRIGger:STATe:CEDGe? <SearchName>

Sets the trigger edge of the clock signal.

Parameters:

<ClockEdge> POSitive | NEGative | EITHer

*RST: POSitive

Parameters for setting and query:

<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:STATe:CLEVel <SearchName>,<ClockLevel>
SEARch:TRIGger:STATe:CLEVel? <SearchName>

Sets the trigger level of the clock signal.

The command has the same effect as with `SEARch:TRIGger:LEVel[:VALue]`.

Parameters:

<ClockLevel>	Range:	-10 to 10
	Increment:	1E-3
	*RST:	0
	Default unit:	V

Parameters for setting and query:

<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:STATe:A[:ENABle] <Searchname>, <State>
SEARch:TRIGger:STATe:A[:ENABle]? <Searchname>
SEARch:TRIGger:STATe:B[:ENABle] <Searchname>, <State>
SEARch:TRIGger:STATe:B[:ENABle]? <Searchname>
SEARch:TRIGger:STATe:C[:ENABle] <Searchname>, <State>
SEARch:TRIGger:STATe:C[:ENABle]? <Searchname>
SEARch:TRIGger:STATe:D[:ENABle] <Searchname>, <State>
SEARch:TRIGger:STATe:D[:ENABle]? <Searchname>

Enables the channel to be considered in the state search. You can enable all channel signals except for the trigger source.

- A[:ENABle]: CH1
- B[:ENABle]: CH2
- C[:ENABle]: CH3
- D[:ENABle]: CH4

Digital channels are not available.

Parameters:

<State> ON | OFF

Parameters for setting and query:

<Searchname> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:STATe:A:LOGic <Searchname>, <Operator>
SEARch:TRIGger:STATe:A:LOGic? <Searchname>
SEARch:TRIGger:STATe:B:LOGic <Searchname>, <Operator>
SEARch:TRIGger:STATe:B:LOGic? <Searchname>
SEARch:TRIGger:STATe:C:LOGic <Searchname>, <Operator>
SEARch:TRIGger:STATe:C:LOGic? <Searchname>
SEARch:TRIGger:STATe:D:LOGic <Searchname>, <Operator>
SEARch:TRIGger:STATe:D:LOGic? <Searchname>

Defines the logic for the indicated channel:

- A: CH1
- B: CH2
- C: CH3

- D: CH4

Parameters:

<Operator> DIRect | HIGH | NOT | LOW

DIRect
Input value remains unchanged

NOT
Input value is inverted

Parameters for setting and query:

<Searchname> String with the name of the search

Usage: Asynchronous command

SEARch:TRIGger:STATe:AB:LOGic <Searchname>, <Operator>
SEARch:TRIGger:STATe:AB:LOGic? <Searchname>
SEARch:TRIGger:STATe:CD:LOGic <Searchname>, <Operator>
SEARch:TRIGger:STATe:CD:LOGic? <Searchname>
SEARch:TRIGger:STATe:ABCD:LOGic <Searchname>, <Operator>
SEARch:TRIGger:STATe:ABCD:LOGic? <Searchname>

Defines the logical combination of the indicated channels after evaluating the previous logical operations:

- AB: CH1 and CH2
- CD: CH3 and CH4
- ABCD: result of AB and CD

Parameters:

<Operator> AND | NAND | OR | NOR

AND: logical AND, conjunctive combination
NAND:logical NOT AND
OR:logical OR, disjunctive combination
NOR: logical NOT OR

Parameters for setting and query:

<Searchname> String with the name of the search

Usage: Asynchronous command

24.15.14 Search on spectrum

CURSor<m>:PEXCursion <Value>

Defines the minimum level by which the waveform must rise or fall so that it will be identified as a maximum or a minimum by the search functions.

Suffix:

<m> The suffix is irrelevant

Parameters:

<Value>	Range:	0 to 100
	Increment:	1
	*RST:	5
	Default unit:	dB

Usage: Asynchronous command

CURSor<m>:THReshold <Value>

Defines an absolute threshold as an additional condition for the peak search. Only peaks that exceed the threshold are detected.

Suffix:
<m> The suffix is irrelevant.

Parameters:
<Value> Threshold in dBm

Usage: Asynchronous command

Firmware/software: Version 2.70

24.15.15 Search gate settings

SEARch:GATE[:STATe]....1631
SEARch:GATE:MODE....1632
SEARch:GATE:SHOW....1632
SEARch:GATE:ABSolute:STARt....1632
SEARch:GATE:ABSolute:STOP....1632
SEARch:GATE:RELative:STARt....1633
SEARch:GATE:RELative:STOP....1633
SEARch:GATE:ZCOupling....1633
SEARch:GATE:ZDIagram....1634

SEARch:GATE[:STATe] <SearchName>,<State>
SEARch:GATE[:STATe]? <SearchName>

Performs the search only on the defined gate area of the source waveform.

Parameters:

<State>	ON | OFF	
	*RST:	OFF

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:GATE:MODE <SearchName>,<Mode>
SEARch:GATE:MODE? <SearchName>

Defines whether the gate settings are configured using absolute or relative values.

Parameters:
<Mode> ABS | REL
*RST: ABS

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:GATE:SHOW <SearchName>,<DisplayState>
SEARch:GATE:SHOW? <SearchName>

If enabled, the gate area is indicated in the source diagram.

Parameters:
<DisplayState> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:GATE:ABSolute:STARt <SearchName>,<Start>
SEARch:GATE:ABSolute:STARt? <SearchName>

Defines the starting value for the gate.

Parameters:
<Start> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:GATE:ABSolute:STOP <SearchName>,<Stop>
SEARch:GATE:ABSolute:STOP? <SearchName>

Defines the end value for the gate.

Parameters:
<Stop> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:GATE:RELative:STARt <SearchName>,<RelativeStart>
SEARch:GATE:RELative:STARt? <SearchName>

Defines the starting value for the gate.

Parameters:
<RelativeStart> Range: 0 to 100
Increment: 0.1
*RST: 0
Default unit: %

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:GATE:RELative:STOP <SearchName>,<RelativeStop>
SEARch:GATE:RELative:STOP? <SearchName>

Defines the end value for the gate.

Parameters:
<RelativeStop> Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:GATE:ZCOupling <SearchName>,<ZoomCoupling>
SEARch:GATE:ZCOupling? <SearchName>

If enabled, the gate area is set to the limits of a zoom area.

The zoom diagramm is selected using `SEARch:GATE:ZDIagram`.

Parameters:
<ZoomCoupling> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName> String with the name of the search

Usage: Asynchronous command

SEARch:GATE:ZDIagram <SearchName>, <DiagramName>, <ZoomName>
SEARch:GATE:ZDIagram? <SearchName>

Selects the zoom to which the gate area is set if `SEARch:GATE:ZCOupling` is set to "ON".

Parameters:
<ZoomDiagram> String with the name of the diagram, where the zoom is defined

Parameters for setting and query:
<SearchName> String with the name of the search

Example: Prerequisite: Search1 and Zoom2 are available.

```
SEARch:GATE:ZCOupling 'Search1',ON
SEARch:GATE:ZDIagram 'Search1','Diagram1','Zoom2'
SEARch:GATE:ZDIagram? 'Search1'
<-- Diagram1;Zoom2
SEARch:GATE:STATe 'Search1', ON
SEARch:GATE:SHOW 'Search2', ON
```

Enables the zoom coupling to define the gate,and selects Zoom2 as gate area. Zoom2 is based on Diagram1. Then the gate is activated and displayed.

Usage: Asynchronous command

24.15.16 Search results

SEARch:RESDiagram:HORZ:ABSolute:POSition 1634
SEARch:RESDiagram:HORZ:ABSolute:SPAN 1635
SEARch:RESDiagram:HORZ:MODE 1635
SEARch:RESDiagram:HORZ:RELative:POSition 1635
SEARch:RESDiagram:HORZ:RELative:SPAN 1636
SEARch:RESDiagram:SHOW 1636
SEARch:RESDiagram:VERT:ABSolute:POSition 1636
SEARch:RESDiagram:VERT:ABSolute:SPAN 1636
SEARch:RESDiagram:VERT:MODE 1637
SEARch:RESDiagram:VERT:RELative:POSition 1637
SEARch:RESDiagram:VERT:RELative:SPAN 1637
SEARch:RESult:LIMit 1638
SEARch:RESult:SHOW 1638
SEARch:RESult:SORT:ASCending 1638
SEARch:RESult:SORT[:MODE] 1639
SEARch:RESult[:ALL]? 1639

SEARch:RESDiagram:HORZ:ABSolute:POSition <SearchName>,<Position>
SEARch:RESDiagram:HORZ:ABSolute:POSition? <SearchName>

Defines the x-value of the centerpoint of the zoom area.

Parameters:

<Position>	Range:	-100E+24 to 100E+24
	Increment:	0.01
	*RST:	0.01

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:HORZ:ABSolute:SPAN <SearchName>,<Span>
SEARch:RESDiagram:HORZ:ABSolute:SPAN? <SearchName>

Defines the width of the zoom area.

Parameters:

<Span>	Range:	0 to 100E+24
	Increment:	0.01
	*RST:	0.01

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:HORZ:MODE <SearchName>,<Mode>
SEARch:RESDiagram:HORZ:MODE? <SearchName>

Defines whether absolute or relative values are used to specify the x-axis values.

Parameters:

<Mode>	ABS | REL	
	*RST:	ABS

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:HORZ:RELative:POSition <SearchName>,<RelPosi>
SEARch:RESDiagram:HORZ:RELative:POSition? <SearchName>

Defines the x-value of the centerpoint of the zoom area.

Parameters:

<RelPosi>	Range:	0 to 100
	Increment:	0.1
	*RST:	100
	Default unit:	%

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:HORZ:RELative:SPAN <SearchName>,<RelativeSpan>
SEARch:RESDiagram:HORZ:RELative:SPAN? <SearchName>

Defines the width of the zoom area.

Parameters:

<RelativeSpan>	Range:	1E-15 to 100
	Increment:	0.1
	*RST:	1
	Default unit:	%

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:SHOW <SearchName>,<ShwSearchWind>
SEARch:RESDiagram:SHOW? <SearchName>

If enabled, a zoom window is displayed for the currently selected search result. The zoom area is indicated in the diagram that displays the source waveform of the search.

Parameters:

<ShwSearchWind>	ON | OFF	
	*RST:	OFF

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:VERT:ABSolute:POSition <SearchName>,<Position>
SEARch:RESDiagram:VERT:ABSolute:POSition? <SearchName>

Defines the y-value of the centerpoint of the zoom area.

Parameters:

<Position>	Range:	-100E+24 to 100E+24
	Increment:	0.01
	*RST:	0.01

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:VERT:ABSolute:SPAN <SearchName>,<Span>
SEARch:RESDiagram:VERT:ABSolute:SPAN? <SearchName>

Defines the height of the zoom area.

Parameters:

<Span>	Range:	0 to 100E+24
	Increment:	0.01
	*RST:	0.01

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:VERT:MODE <SearchName>,<Mode>
SEARch:RESDiagram:VERT:MODE? <SearchName>

Defines whether absolute or relative values are used to specify the y-axis values.

Parameters:

<Mode>	ABS | REL	
	*RST:	ABS

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:VERT:RELative:POSition <SearchName>,<RelPosi>
SEARch:RESDiagram:VERT:RELative:POSition? <SearchName>

Defines the y-value of the centerpoint of the zoom area.

Parameters:

<RelPosi>	Range:	0 to 100
	Increment:	0.1
	*RST:	100
	Default unit:	%

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESDiagram:VERT:RELative:SPAN <SearchName>,<RelativeSpan>
SEARch:RESDiagram:VERT:RELative:SPAN? <SearchName>

Defines the height of the zoom area.

Parameters:

<RelativeSpan>	Range:	1E-15 to 100
	Increment:	0.1
	*RST:	1
	Default unit:	%

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESult:LIMit <SearchName>,<ResultListLimit>
SEARch:RESult:LIMit? <SearchName>

Defines the maximum number of entries in the search result table.

Parameters:
<ResultListLimit> Range: 1 to 1000
Increment: 1
*RST: 100

Parameters for setting and query:
<SearchName> Search definition

Example: See Chapter 24.3.4.1, "Searching for a pulse of specified width", on page 1266

Usage: Asynchronous command

SEARch:RESult:SHOW <SearchName>,<ShowResultTable>
SEARch:RESult:SHOW? <SearchName>

Displays or hides the search result table.

Parameters:
<ShowResultTable> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESult:SORT:ASCending <SearchName>,<SortAscending>
SEARch:RESult:SORT:ASCending? <SearchName>

If enabled, the results are listed in asscending order, i.e. the smallest value at the top.

Parameters:
<SortAscending> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESult:SORT[:MODE] <SearchName>,<SortMode>
SEARch:RESult:SORT[:MODE]? <SearchName>

Sorts the search result table by x-value position or value of the result.

Parameters:

<SortMode> POSition | VALue

POSition
Sorts the search result table by the x-value position.

VALue
Sorts the search result table by the value of the result.

*RST: POSition

Parameters for setting and query:

<SearchName> Search definition

Usage: Asynchronous command

SEARch:RESult[:ALL]? <SearchName>

Returns all search results.

Query parameters:

<SearchName> Search definition

Return values:

<Data> List of search results, separated by commas. For each result, six values are returned:
1. Acquisition index, currently always 0.
2. X-position of the search result
3. Y-position of the search result, currently not relevant
4. Type of the search result (Edge, Glitch, ...)
5. Slope or polarity of the search result
6. For runt, glitch, width, and window searches, the value contains the width. For timeout and interval searches, it contains the timeout. For transition searches, it contains the slew rate. For all other searches, the value is not relevant.
If a value is not relevant, 9.91E+37 is returned.

Example:

```
SEAR:RES? 'Search1'
0,1.5375e-007,-84,Edge,Positive,9.91E+37,
0,5.3e-008,-84,Edge,Positive,9.91E+37
```

The query returns two search results for edge search on rising edges at X-position 153,75 ns and 53 ns.

Usage: Query only
Asynchronous command

24.16 Data management

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

- Instrument settings....1640
- Savesets and one file....1648
- Waveform data transmission....1649
- Waveform data export to file....1649
- Waveform histogram export to file....1656
- Eye....1657
- Long term measurement results and measurement histogram export to file....1659
- Screenshots....1660
- Reports....1664
- Autonaming....1666
- CSV export....1668
- External application....1668

24.16.1 Instrument settings

The Mass MEMory subsystem provides commands to access the storage media and to save and reload instrument settings.

File and directory names

The <file_name> and <directory_name> parameters are strings. Some commands use a fixed directory; for others the <file_name> can contain the complete path including the drive name and all subdirectories. If no complete path is specified, the file location is relative to the current directory, queried with `MMEMory:CDIRectory?`. The file name itself may contain the period as a separator for extensions.

Example: '`C:\TEMP\TRASH\test.txt`' for the file named `test.txt` in the `TEMP\TRASH` subdirectory of the internal hard disk drive C:\.

File and directory names can be chosen according to Windows™ conventions; the restrictions placed on file names known from DOS systems do not apply. All letters and numbers are allowed, as well as the special characters "_", "^", "$", "~", "!", "#", "%", "&", "-", "{", "}", "(", ")", "@" and "`". Reserved file names are CON, AUX, COM1, ..., COM4, LPT1, ..., LPT3, NUL and PRN.

The use of wildcards ? and * is not allowed.

MMEMory:DRIVes?....1641
MMEMory:MSIS....1641
MMEMory:DCATalog?....1641
MMEMory:DCATalog:LENGth?....1642
MMEMory:CDIRectory....1642
MMEMory:MDIRectory....1642
MMEMory:RDIRectory....1643
MMEMory:CATalog?....1643
MMEMory:CATalog:LENGth?....1644
MMEMory:COPY....1644
MMEMory:MOVE....1644
MMEMory:DELete....1645
MMEMory:DATA....1645
MMEMory:ATTRibute....1646
MMEMory:SAV....1646
MMEMory:RCL....1646
MMEMory:STORe:STATe....1647
MMEMory:LOAD:STATe....1647

MMEMory:DRIVes?

Returns a list of the logical drives of the instrument as configured in the operating system.

Return values:

<Drive> List of strings, for example, `"C:\","F:\","H:\"`

Usage: Query only

MMEMory:MSIS [<msus>]

Changes the default storage device to the indicated drive or network server.

Parameters:

<msus> String parameter. Drives are indicated with their drive letter, network servers require the UNC format.

Example: `MMEM:MSIS 'C:'`

Example: `MMEM:MSIS '\\server1\share1'`

MMEMory:DCATalog? [<PathName>]

Returns the subdirectories of the current or of a specified directory.

Query parameters:

<PathName> String parameter to specify the directory. If the directory is omitted, the command queries the contents of the current directory, to be set and queried with `MMEMory:CDIRectory`.

Return values:

<FileEntry> Names of the subdirectories separated by colons. The first two strings are related to the parent directory.

Example:

```
MMEM:DCAT?
".",".","Documents and Settings","Program
Files","temp"
```

Usage: Query only

MMEMory:DCATalog:LENGth? [<PathName>]

Returns the number of subdirectories of the current or of a specified directory. The number includes the parent directory strings "." and ".." and corresponds to the number of strings returned by the `MMEMory:DCATalog?` command.

Query parameters:

<PathName> String parameter to specify the directory. If the directory is omitted, the command queries the contents of the current directory, to be set and queried with `MMEMory:CDIRectory`.

Return values:

<FileEntryCount> Number of parent and subdirectories.

Example:

```
MMEM:DCAT:LENG?
5
```

Usage: Query only

MMEMory:CDIRectory [<DirectoryName>]

Changes the default directory for file access.

Parameters:

<DirectoryName> String parameter to specify the directory.
If the string also contains a drive letter or network server name, the command `MMEMory:MSIS` is executed implicitely.

*RST: "\"

Example:

```
MMEM:CDIR 'C:\USER\DATA'
```

Usage: SCPI confirmed

MMEMory:MDIRectory <DirectoryName>

Creates a new directory with the specified name.

Setting parameters:

<DirectoryName> String parameter to specify the new directory. If the path consists of several subdirectories, the complete tree will be created if necessary.
If no drive letter or server name is indicated, the directory is created on the default storage device specified with `MMEMory:MSIS`.

Example: `MMEM:MDIR 'C:\USER\DATA'`

Usage: Setting only

MMEMory:RDIRectory <DirectoryName>

Deletes the specified directory.

Setting parameters:

<DirectoryName> String parameter to specify the directory to be deleted.
This directory must be empty, otherwise it is not deleted.

Example: `MMEM:RDIR 'C:\USER\TEST'`

Usage: Setting only

MMEMory:CATalog? [<PathName>][, <Format>]

Returns the a list of files contained in the specified directory. The result corresponds to the number of files returned by the MMEMory:CATalog:LENgth command.

Query parameters:

<PathName> String parameter to specify the directory. If the directory is omitted, the command queries directory specified with `MMEMory:CDIRectory`.

<Format> ALL | WTIMe

ALL: Extended result including file, date, time and attributes
WTIMe: Extended result including file, date, time

Return values:

<UsedMemory> Total amount of storage currently used in the directory, in bytes.

<FreeMemory> Total amount of storage available in the directory, in bytes.

<FileEntry> All files and subdirectories of the directory are listed with their file name, format and size in bytes. The first two strings are related to the parent directory.

Example:

```
MMEM:CAT? 'C:\USER\DATA'
529479,1831777894400,".,DIR,0",".,DIR,0",
"Backup,DIR,0","CSS,DIR,0","DATEN,DIR,0",
"Commands.jar,BIN,529479","FAVORITES,DIR,0",
"LOG,DIR,0","DATA,DIR,0","test,DIR,0",
"TotalCMD,DIR,0"
```

Usage: Query only
SCPI confirmed

MMEMory:CATalog:LENGth? [<PathName>]

Returns the number of files and subdirectories of the current or specified directory. The number includes the parent directory strings "." and ".." and it corresponds to the number of <FileEntry> strings returned by the `MMEMory:CATalog?` command.

Query parameters:

<PathName> String parameter, directory to be queried. If the directory is omitted, the current directory is queried, specified with `MMEMory:CDIRectory`.

Return values:

<Count> Number of files and subdirectories including parent directory entries.

Example:

```
MMEM:CDIR 'C:\USER\DATA'
MMEM:CAT:LENG?
11
```

Usage: Query only

MMEMory:COPY <FileSource>[, <FileDestination>]

Copies an existing file to a new file.

Setting parameters:

<FileSource> String parameter, contains name and path of the file to be copied. Wildcards (* and ?) are allowed.

<FileDestination> String parameter, contains name and path of the new file. If the file already exists, it is overwritten without notice. If no file destination is specified, the source file is written to the current directory specified with `MMEMory:CDIRectory`.

Example:

```
MMEM:COPY 'C:\Users\Public\Documents
\Rohde-Schwarz\RTx\RefWaveforms
\RefCurve_2011-03-16*.bin','E:'
```

Copies all reference waveforms saved on March 16, 2011 to an external storage medium, mapped to drive `E:\`.

Usage: Setting only
SCPI confirmed

MMEMory:MOVE <FileSource>, <FileDestination>

Moves the specified file to a new location on the same drive and renames it.

Setting parameters:

<FileSource> String parameter, contains name and path of the file to be copied.
Wildcards (* and ?) are allowed. Therefore, specify a directory for <FileDestination>. Renaming is not possible.

<FileDestination> String parameter, contains name and path of the new file. If no path is specified, the <FileSource> directory is used - the file is renamed.

Example: `MMEM:MOVE 'C:\USER\DATA\SETUP.CFG','C:\STORE'`
Moves the file "Setup.cfg" from the directory `C:\USER\DATA` to `C:\STORE`.

Usage: Setting only
SCPI confirmed

MMEMory:DELete <FileName>

Removes the specified file(s). To delete directories, use `MMEMory:RDIRectory`.

Setting parameters:

<FileName> String parameter to specify the name and directory of the file to be removed. Wildcards (* and ?) are allowed.
If no path is defined, the current directory is used, specified with `MMEMory:CDIRectory`.

Example: `MMEM:DEL '*.CFG'`
Deletes all cfg files from the current directory.

Usage: Setting only
SCPI confirmed

MMEMory:DATA <FileName>, <Data>
MMEMory:DATA? <FileName>

Stores data in the specified file to the storage location specified using `MMEMory:CDIRectory`.

Parameters:

<Data> <block>

488.2 block data format. The delimiter EOI must be selected to achieve correct data transfer.
The block begins with character '#'. The next digit is the length of the length information, followed by this given number of digits providing the number of bytes in the following binary data.

Parameters for setting and query:

<FileName> String parameter, the name of the file the data is stored to.

Example: `MMEM:DATA 'abc.txt', #216This is the file`
#2: the length infomation has two digits
16: the binary data has 16 bytes

Example: `MMEM:DATA? 'abc.txt'`
Returns the data from file `abc.txt`.

MMEMory:ATTRibute <FileName>, <Attributes>
MMEMory:ATTRibute? <FileName>

Sets file attributes for the specified file(s). The command can be used for files only.

Setting parameters:

<Attributes> String with attributes and setting information.
'+' before the attribute: sets the attribute
'-' before the attribute: deletes the attribute
'R': read only
'A': archive file
'S': system file
'H': hidden file

Parameters for setting and query:

<FileName> String parameter, contains name and path of the file. Wildcards (* and ?) are allowed.

Return values:

<FileEntry> String containing: "<file_name>,<file_attributes>"

Example: `MMEM:ATTR 'C:\USER\DATA\*.LOG', '-R -A'`
Deletes the read-only and archive attributes from all LOG files in the directory`C:\USER\DATA\*.LOG`.

Example:
```
MMEM:ATTR? 'C:\USER\DATA\*.*'
"Datei1.LOG,A", "Datei2.LOG,A",
"Datei3.LOG,ASH", "Datei4.DLL,RSH",
"Datei5.INI,SH"
```

MMEMory:SAV <FileDestination>

Stores the current instrument settings to the specified file.

This command has the same effect as the combination of `*SAV` and `MMEMory:STORe:STATe`.

Parameters:

<FileDestination> String parameter specifying path and filename of the target file. Wildcards are not allowed.

Example: `MMEM:SAV 'C:\mysavefile.dfl'`
Saves the current instrument settings to the file `mysavefile.dfl` located in the directory C:\.

Usage: Event

MMEMory:RCL <FileSource>

Restores the instrument settings from the specified file.

This command has the same effect as the combination of `MMEMory:LOAD:STATe` and `*RCL`.

Parameters:

'<FileSource>' String parameter specifying the path and filename of the source file. Wildcards are not allowed.

Example: `MMEM:RCL 'C:\mysavefile.dfl'`
Loads and activates the instrument settings from the file `mysavefile.dfl` located in the directory C:\.

Usage: Event

MMEMory:STORe:STATe <MemoryNumber>, <FileName>

Stores the instrument settings from the specified internal memory to the specified file. To store the current instrument settings to the internal memory, use `*SAV` first.

Setting parameters:

<MemoryNumber> Number of the internal memory

Range: 1 to 99

<FileName> String parameter specifying the complete path and filename of the source file.

Example:
```
*SAV 4
MMEM:STORe:STATe 4,'C:
\Settings\Settings_1051.dfl'
```
Saves current instrument settings to the internal memory number 4. Then stores the settings from the internal memory number 4 to the file `C:\Settings\Settings_1051.dfl`.

Usage: Setting only

MMEMory:LOAD:STATe <MemoryNumber>, <FileName>

Loads the instrument settings from the specified file to the specified internal memory. After the file has been loaded, the settings must be activated using a `*RCL` command.

Setting parameters:

<MemoryNumber> Number of the internal memory

Range: 1 to 99

<FileName> String parameter specifying the complete path and filename of the source file.

Example:
```
MMEM:LOAD:STATe 4,'C:
\Settings\Settings_1051.dfl'
*RCL 4
```
Loads instrument settings from the file `C:\Settings\Settings_1051.dfl` to the internal memory number 4, and then activates the settings in internal memory number 4.

Usage: Setting only

24.16.2 Savesets and one file

SAVeset:CONFig:PREView....1648
SAVeset:ONEFile:NAME....1648
SAVeset:ONEFile:SAVE....1648
SAVeset:ONEFile:OPEN....1648
SAVeset:ONEFile:USERpref....1648

SAVeset:CONFig:PREView <Include>

If set to OFF, the saveset is stored without the preview image to reduce the file size.

Use the command each time before you save a saveset.

Parameters:
<Include> ON | OFF
*RST: ON

Usage: Asynchronous command

SAVeset:ONEFile:NAME <Name>

Sets the path and the file name of the One File. The file format is ZIP.

Parameters:
<Name> String parameter

Usage: Asynchronous command

SAVeset:ONEFile:SAVE

Usage: Event

Saves the One File data to the file that is specified with `SAVeset:ONEFile:NAME`.

SAVeset:ONEFile:OPEN

Loads the One File that is specified with `SAVeset:ONEFile:NAME`.

Usage: Event

SAVeset:ONEFile:USERpref <Include>

If ON, the user preferences settings are also restored together with the saveset and reference waveform. User preferences settings include the user-specific display settings like diagram layout, toolbar, intensity and transparency settings.

Parameters:
<Include> ON | OFF
*RST: OFF

Usage: Asynchronous command

24.16.3 Waveform data transmission

The R&S RTO6 provides specific data export commands for the various waveform types. The commands transmit the data of the waveform points from the instrument to the controlling computer. The data can be used in MATLAB, for example.

The commands are described in the relevant chapters:

- Analog waveforms: Chapter 24.8.6, "Waveform data", on page 1342
- Reference waveforms: Chapter 24.10.2.3, "Waveform data export", on page 1453
- Math waveforms: Chapter 24.10.3, "Mathematics", on page 1456
- Spectrum waveforms: Chapter 24.13.2, "Waveform data", on page 1578
- Logic channels: Chapter 24.18.5, "MSO data", on page 2440
- I/Q data: Chapter 24.21.2, "I/Q data output", on page 2477

24.16.4 Waveform data export to file

The resulting files of waveforms exports are described in Chapter 12.2.1, "Waveform export files", on page 453.

EXPort:WAVeform:SOURce 1649
EXPort:WAVeform:MULTichannel 1650
CHANnel<m>:EXPortstate 1651
EXPort:WAVeform:NAME 1651
EXPort:WAVeform:SAVE 1651
EXPort:WAVeform:SCOPe 1652
EXPort:WAVeform:STARt 1652
EXPort:WAVeform:STOP 1652
EXPort:WAVeform:ZOOM 1653
EXPort:WAVeform:CURSorset 1653
EXPort:WAVeform:MEAS 1653
EXPort:WAVeform:DLOGging 1654
EXPort:WAVeform:TIMestamps 1654
EXPort:WAVeform:INCXvalues 1655
EXPort:WAVeform:RAW 1655
EXPort:WAVeform:DISPlayoff 1656
EXPort:WAVeform:FASTexport 1656

EXPort:WAVeform:SOURce <Source>

Selects the waveform to be exported to file.

The commands takes effect if `EXPort:WAVeform:MULTichannel` is `OFF`.

Parameters:

<Source> | C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | MSOB1 | MSOB2 | MSOB3 | MSOB4 | TRK1 | TRK2 | TRK3 | TRK4 | TRK5 | TRK6 | TRK7 | TRK8 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2 | BATHtub1 | BMEas1 | SRESponse1 | SRHist1 | SDHist1 | TJHist1 | DJHist1 | DDJHist1 | PJHist1 | RJOHist1 | SRTRack1 | SDTRack1 | TJTRack1 | DJTRack1 | DDJTrack1 | PJTRack1 | RJOTrack1 | TJSPectrum1 | DDJSpectrum1 | PJSPectrum1 | RJOSpectrum1 | SEYE1 | BATHtub2 | BMEas2 | SRESponse2 | SRHist2 | SDHist2 | TJHist2 | DJHist2 | DDJHist2 | PJHist2 | RJOHist2 | SRTRack2 | SDTRack2 | TJTRack2 | DJTRack2 | DDJTrack2 | PJTRack2 | RJOTrack2 | TJSPectrum2 | DDJSpectrum2 | PJSPectrum2 | RJOSpectrum2 | SEYE2 | NBAThtub1 | NBMeas1 | TNHist1 | DNHist1 | DDNHist1 | PNHist1 | RNOHist1 | TNTRack1 | DNTRack1 | DDNTrack1 | PNTRack1 | RNOTrack1 | TNSPectrum1 | DDNSpectrum1 | PNSPectrum1 | RNOSpectrum1 | NBAThtub2 | NBMeas2 | TNHist2 | DNHist2 | DDNHist2 | PNHist2 | RNOHist2 | TNTRack2 | DNTRack2 | DDNTrack2 | PNTRack2 | RNOTrack2 | TNSPectrum2 | DDNSpectrum2 | PNSPectrum2 | RNOSpectrum2 | PJDHist1 | BUJHist1 | RJHist1 | PJDTrack1 | PJDSpectrum1 | PJDHist2 | BUJHist2 | RJHist2 | PJDTrack2 | PJDSpectrum2 | PNDHist1 | BUNHist1 | RNHist1 | PNDTrack1 | PNDSpectrum1 | PNDHist2 | BUNHist2 | RNHist2 | PNDTrack2 | PNDSpectrum2

*RST: C1W1

Example: See Chapter 24.3.5.2, "Exporting waveform data to file", on page 1267

Usage: Asynchronous command

EXPort:WAVeform:MULTichannel <MltChXpt>

Enables or disables the multichannel export.

If you enable the multichannel export, all active channels are included to the export data. You can change the export state using the `CHANnel<m>:EXPortstate` command.

If multichannel export is disabled, select the waveform to be exported using the `EXPort:WAVeform:SOURce` command.

Note that `CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?` returns the data of all channels that are selected for export, no matter of the channel suffix.

Parameters:
<MltChXpt> ON | OFF
*RST: OFF

Usage: Asynchronous command

CHANnel<m>:EXPortstate <ExportState>

Includes or excludes the indicated channel in waveform export. The data of channel waveform 1 is exported.

The commands takes effect if `EXPort:WAVeform:MULTichannel` is `ON`.

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<ExportState> ON | OFF
If you enable the multichannel export, the export state of all active channels is automatically set ON.
*RST: OFF

Usage: Asynchronous command

EXPort:WAVeform:NAME <FileName>

Sets the file name, file format and path to save the waveform to.

See also: Chapter 12.2.1, "Waveform export files", on page 453

Parameters:
<FileName> String with path and file name with extension .xml, .bin, or .csv

Example:
```
EXPort:WAVeform:NAME 'C:\temp\Export_Chl1.xml'
EXPort:WAVeform:SAVE
```
Saves the waveform data in XML format to `C:\temp\Export_Ch1.xml`.

Example:
```
EXPort:WAVeform:NAME 'C:\temp\Export_Ch2.bin'
EXPort:WAVeform:SAVE
```
Saves the waveform data in binary format to `C:\temp\Export_Ch2.bin`.

Usage: Asynchronous command

EXPort:WAVeform:SAVE

Saves the waveform(s) to the file specified with `EXPort:WAVeform:NAME`. The file format is also set using the `...NAME` command.

Example: See Chapter 24.3.5.2, "Exporting waveform data to file", on page 1267

Usage: Event

EXPort:WAVeform:SCOPe <Scope>

Defines the part of the waveform record that has to be stored.

Parameters:

<Scope> WFM | ZOOM | CURSor | GATE | MANual

WFM
Complete waveform

ZOOM
Data included in the zoom area if a zoom is defined for the source waveform.

CURSor
Data between the cursor lines if a cursor measurement is defined for the source waveform.

GATE
data included in the measurement gate if a gated measurement is defined for the source waveform.

MANual
Saves the data between user-defined start and stop values to be set with `EXPort:WAVeform:STARt` and `EXPort:WAVeform:STOP`.

*RST: WFM

Example: See Chapter 24.3.5.2, "Exporting waveform data to file", on page 1267

Usage: Asynchronous command

EXPort:WAVeform:STARt <Start>

Sets the start value of the waveform section for export, if `EXPort:WAVeform:SCOPe` is set to `Manual`.

Parameters:

<Start>
Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01
Default unit: s

Usage: Asynchronous command

EXPort:WAVeform:STOP <Stop>

Sets the end value of the waveform section for export, if `EXPort:WAVeform:SCOPe` is set to `Manual`.

Parameters:

<Stop>

Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01
Default unit: s

Usage: Asynchronous command

EXPort:WAVeform:ZOOM <ZoomDiagram>

Sets the zoom area to be used for limited data export if `EXPort:WAVeform:SCOPe` is set to `ZOOM`.

Parameters:

<ZoomDiagram> Name of the diagram on which the zoom area is based.

Example: See "Exporting interleaved x/y data of a zoom to CSV file" on page 1269

Usage: Asynchronous command

EXPort:WAVeform:CURSorset <Cursorset>

Sets the cursor set to be used for limited data export if `EXPort:WAVeform:SCOPe` is set to `CURSor`.

Parameters:

<Cursorset>

CURSOR1 | CURSor1 | CURSOR2 | CURSor2 | CURSOR3 | CURSor3 | CURSOR4 | CURSor4

CURSOR1 = CURSor1, CURSOR2 = CURSor2, CURSOR3 = CURSor3, CURSOR4 = CURSor4

Usage: Asynchronous command

EXPort:WAVeform:MEAS <MeasGate>

Sets the gate to be used for limited data export if `EXPort:WAVeform:SCOPe` is set to `GATE`.

Parameters:

<MeasGate>

MEAS1 | MEAS2 | MEAS3 | MEAS4 | MEAS5 | MEAS6 | MEAS7 | MEAS8

Measurement for which the gate is defined.

Example: See "Exporting raw data of a measurement gate to BIN file" on page 1268

Usage: Asynchronous command

EXPort:WAVeform:DLOGging <DataLogging>

The command enables the export of subsequent acquisitions of the selected waveforms. The waveforms are taken from a running Nx Single acquisition (data logging, history is disabled), or from the history (multiple waveforms, history is enabled).

If the history is disabled (`CHANnel<m>[:WAVeform<n>]:HISTory[:STATe]`) and data logging is enabled, a specified number of waveforms is transferred to file directly during RUN Nx SINGLE acquisition. Enabling data logging stops a running acquisition. Set the number of acquisitions to be acquired and stored with `ACQuire:COUNt` and start export using `RUNSingle`.

If the history is enabled, the subsequent waveforms are taken from the history. Specify the range with `CHANnel<m>[:WAVeform<n>]:HISTory:STARt` and `CHANnel<m>[:WAVeform<n>]:HISTory:STOP`. Then play the history with `CHANnel<m>[:WAVeform<n>]:HISTory:PLAY`.

The commands `EXPort:WAVeform:SAVE`, `CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?` and `CHANnel<m>[:WAVeform<n>]:DATA:HEADer?` are not available if data logging is enabled. The `RUNContinous` command disables data logging.

If data logging is off, and the history is enabled, one waveform out of the history is written to file. Specify the waveform using `CHANnel<m>[:WAVeform<n>]:HISTory:CURRent` and save it using `EXPort:WAVeform:SAVE`.

Parameters:

<DataLogging> ON | OFF

*RST: OFF

Example: See:
"Exporting multiple running acquisitions of a single waveform to XML file" on page 1270
"Exporting multiple acquisition of the history to XML file" on page 1271
"Exporting a single acquisition of the history to BIN file" on page 1270

Usage: Asynchronous command

EXPort:WAVeform:TIMestamps <UseTimestamps>

Exports the relative timestamps of all history waveforms to the waveform data file. The time is written at the beginning of each waveform record.

Parameters:

<UseTimestamps> ON | OFF

*RST: OFF

Usage: Asynchronous command

EXPort:WAVeform:INCXvalues <IncHorValues>

Includes horizontal values in the retrieved data (time or frequency values, depending on the waveform). X and Y-values are written alternately to the file. If disabled, only Y-values - mostly voltage values - are written. The X-values are always returned in 64 bit real format, regardless of the defined data format.

The setting is not available for the export of raw data.

The command affects the content of export files as well as data retrieved with:

- `CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?`
- `CALCulate:MATH<m>:DATA[:VALues]?`
- `REFCurve<m>:DATA[:VALues]?`
- `DIGital<m>:DATA[:VALues]?`
- `BUS<m>:PARallel:DATA[:VALues]?`
- `ZVC:Z<m>:I<n>:DATA[:VALues]?`
- `ZVC:Z<m>:V<n>:DATA[:VALues]?`

Parameters:

<IncHorValues> ON | OFF

*RST: OFF

Example: See:
"Exporting interleaved x/y data of a single waveform to CSV file" on page 1269
"Exporting interleaved x/y data of a zoom to CSV file" on page 1269

Usage: Asynchronous command

EXPort:WAVeform:RAW <RawValues>

Enables the export of raw sample data, and sets the data format to integer 8 bit. In high definition acquisition mode, the data format is integer 16 bit. Fot INT16, you can set the byte order using the `FORMat:BORDer` command.

The raw format reduces the file size but changes also the precision of the values.

Currently, the setting is not available for the export of digital channel data and data of R&S RT-ZVC channels.

The raw export of interleaved X/Y values is not supported.

Parameters:

<RawValues> ON | OFF

*RST: OFF

Example: See:
"Exporting raw data of a single waveform to BIN file" on page 1267
"Exporting raw data of a measurement gate to BIN file" on page 1268

Usage: Asynchronous command

EXPort:WAVeform:DISPlayoff <FastExport>

Enables or disables the display update during an Nx Single acquisition.

Parameters:

<FastExport> ON | OFF

ON: Disables the display update for maximum export speed.
OFF: Enables the display update. The export is slower.

*RST: OFF

Usage: Asynchronous command

EXPort:WAVeform:FASTexport <Enable>

To improve the performance of data export to file, the measurements are performed slower while the data export speeds up.

Setting parameters:

<Enable> ON | OFF

Example: See Chapter 24.3.5.2, "Exporting waveform data to file", on page 1267

Usage: Setting only
Asynchronous command

24.16.5 Waveform histogram export to file

EXPort:HISTogram:SELect....1656
EXPort:HISTogram:INCidence....1656
EXPort:HISTogram:NAME....1657
EXPort:HISTogram:SAVE....1657
EXPort:HISTogram:DATA?....1657

EXPort:HISTogram:SELect <Name>

Selects the histogram to be exported.

Parameters:

<Name> String with the histogram name.

Example: See "Exporting histogram data to file" on page 1264

Usage: Asynchronous command

EXPort:HISTogram:INCidence <Incidence>

Sets the mode of exported data: relative or absolute frequency of amplitude values.

Parameters:
<Incidence> ABS | REL
*RST: REL

Example: See "Exporting histogram data to file" on page 1264

Usage: Asynchronous command

EXPort:HISTogram:NAME <Path>

Sets the file name and path to save the histogram to.

Parameters:
<Path> String with path and file name. The file extension defines the file format: XML, CSV, or BIN.

Example: See "Exporting histogram data to file" on page 1264

Usage: Asynchronous command

EXPort:HISTogram:SAVE

Saves the histogram to the file specified with `EXPort:HISTogram:NAME`.

Example: See "Exporting histogram data to file" on page 1264

Usage: Event
Asynchronous command

EXPort:HISTogram:DATA?

Transfers the histogram data to the controlling computer. The data can be used in MATLAB, for example.

To set the export data format, use `FORMat[:DATA]`.

Return values:
<Data> List of values according to the format settings and `EXPort:HISTogram:INCidence`.

Example: See "Transferring histogram data" on page 1264.

Usage: Query only

24.16.6 Eye

EXPort:EYE:DATA?.....1658
EXPort:EYE:NAME.....1658
EXPort:EYE:SAVE.....1658
EXPort:EYE:SELect.....1658
EXPort:EYE:SOURce?.....1658

EXPort:EYE:DATA?

Transfers the eye data to the controlling computer. The data can be used in MATLAB, for example.

To set the export data format, use `FORMat[:DATA]`.

Usage: Query only

EXPort:EYE:NAME <Path>

Sets the path, the file name, and the file format for the eye file. Available file formats are *csv and *bin.

Parameters:
<Path>

Usage: Asynchronous command

EXPort:EYE:SAVE

Saves the selected eye to the file that is specified with `EXPort:EYE:NAME`.

Usage: Event

EXPort:EYE:SELect <SelectableEye>

Selects the eye that you want to save to a file.

Parameters:
<SelectableEye> EYE1 | EYE2 | EYE3 | EYE4
*RST: EYE1

Usage: Asynchronous command

EXPort:EYE:SOURce?

Returns the source of the eye measurement, selected with `EXPort:EYE:SELect`.

Return values:
<Source> C1W1 | C1W2 | C1W3 | C1W4 | C2W1 | C2W2 | C2W3 | C2W4 | C3W1 | C3W2 | C3W3 | C3W4 | C4W1 | C4W2 | C4W3 | C4W4
*RST: NONE

Usage: Query only
Asynchronous command

24.16.7 Long term measurement results and measurement histogram export to file

EXPort:MEASurement:SELect....1659
EXPort:MEASurement:TYPE....1659
EXPort:MEASurement:NAME....1659
EXPort:MEASurement:SAVE....1660
EXPort:MEASurement:DATA?....1660

EXPort:MEASurement:SELect <SelcMeas>

Selects the measurement for export of long term or measurement histogram data.

Parameters:

<SelcMeas> MEAS1 | MEAS2 | MEAS3 | MEAS4 | MEAS5 | MEAS6 | MEAS7 | MEAS8

*RST: MEAS1

Example: See "Exporting long-term measurement data to file" on page 1264

Usage: Asynchronous command

EXPort:MEASurement:TYPE <ExportType>

You can export the result data of the long term measurement, or the measurement histogram, or the track data.

To export the measurement histogram, it must be enabled using `MEASurement<m>:STATistics:HISTogram`.

To export the long term results, the long term measurement must be enabled using `MEASurement<m>:LTMeas[:STATe]`.

To export a track, the track must be enabled before. Track measurements require an option, see "Enable (Track)" on page 361.

Parameters:

<ExportType> LONGterm | HISTogram | TRACk

LONGTERM = LONGterm, HISTOGRAM = HISTogram

*RST: HISTogram

Example: See "Exporting long-term measurement data to file" on page 1264

Usage: Asynchronous command

EXPort:MEASurement:NAME <Path>

Sets the file name and path to save the long term or measurement histogram data to.

Parameters:

<Path> String with path and file name. The file extension defines the file format: XML, CSV, or BIN.

Example: See "Exporting long-term measurement data to file" on page 1264

Usage: Asynchronous command

EXPort:MEASurement:SAVE

Saves the long term or measurement histogram results to the file specified using `EXPort:MEASurement:NAME`.

The measurement data can be exported as absolute or relative values, which is defined using `EXPort:HISTogram:INCidence`.

Example: See "Exporting long-term measurement data to file" on page 1264

Usage: Event

EXPort:MEASurement:DATA?

Transfers the long term measurement data to the controlling computer. The data can be used in MATLAB, for example. The transferred data are the same as the data written to files.

To set the export data format, use `FORMat[:DATA]`.

Return values:

<Data> List of values according to the format settings
Long term data:
If statistics are disabled, the current value of each long term point is returned.
If statistics are enabled (`MEASurement<m>:STATistics[:ENABle]`), seven values for each long term point are returned: upper peak, lower peak, RMS, standard deviation, average, number of measured results per long term point, number of waveforms per long term point.
For measurement histograms, absolute values are returned.
See also: Chapter 12.2.5, "Result plots analysis", on page 469.

Example: See "Transferring long-term measurement data" on page 1265

Usage: Query only

24.16.8 Screenshots

The HCOPy subsystem and some other commands control the output of display information for documentation purposes. The instrument allows two independent output configurations which can be set separately with the suffix.

Note that the remote mode is intended for maximum performance. Therefore, the display does not follow the remote commmands consistantly. To get a correct screenshot, turn the display on using `SYST:DISP:UPD ON`.

HCOPy:DESTination<1..2>.....1661
MMEMory:NAME.....1661
HCOPy:DEVice<m>:LANGuage.....1662
HCOPy:DEVice<m>:INVerse.....1662
HCOPy:WBKG.....1662
HCOPy:CMAP<m>:DEFault.....1663
HCOPy:SSD.....1663
HCOPy:ISBA.....1663
HCOPy:IMMediate<m>[:DUM].....1664
HCOPy:IMMediate<m>:NEXT.....1664

HCOPy:DESTination<1..2> <Medium>

Selects the output medium: file, printer or clipboard.

Suffix:
1..2 Selects the output configuration.

Parameters:
<Medium> MMEM | SYST:COMM:CLIP

String parameter

MMEM
Directs the display image to a file. The `MMEMory:NAME` command defines the file name. The file format is defined with `HCOPy:DEVice<m>:LANGuage`.

SYST:COMM:CLIP
Directs the hardcopy to the clipboard.

*RST: SYST:COMM:CLIP

Example: `HCOP:DEST 'MMEM'`
See also Chapter 24.3.5.1, "Saving a screenshot to file", on page 1266

Usage: Asynchronous command

MMEMory:NAME <FileName>

Defines the file name when an image of the display is stored to a file.

Setting parameters:
<FileName> String parameter specifiying path and file name of the screenshot.

Example: See Chapter 24.3.5.1, "Saving a screenshot to file", on page 1266

Usage: Setting only
SCPI confirmed
Asynchronous command

HCOPy:DEVice<m>:LANGuage <FileFormat>

Defines the file format for output of the display image to file.

To set the output to file, use `HCOPy:DESTination<1..2>` with parameter `'MMEM'`.

Suffix:
<m> 1..2
Selects the output configuration.

Parameters:
<FileFormat> PNG | JPG | BMP | TIFF | PDF
*RST: PNG

Example: See Chapter 24.3.5.1, "Saving a screenshot to file", on page 1266

Usage: Asynchronous command

HCOPy:DEVice<m>:INVerse <InverseColor>

Inverts the colors of the output, i.e. a dark waveform is printed on a white background. Screenshots are inverted per default when you save using remote commands. Turn inversion off if you want the screen display in the screenshot.

See also:

- `HCOPy:WBKG` on page 1662
- "White background" on page 477

Suffix:
<m> 1..2
Selects the output configuration.

Parameters:
<InverseColor> ON | OFF
*RST: ON

Usage: Asynchronous command

HCOPy:WBKG <WhiteBackground>

Inverts the background color. So you can print waveforms with normal waveform collors on white background.

If both `HCOPy:WBKG` and `HCOPy:DEVice<m>:INVerse` are ON, the instrument inverts the background twice, and it appears black.

See also: "White background" on page 477.

Parameters:
<WhiteBackground> ON | OFF
*RST: OFF

Usage: Asynchronous command

HCOPy:CMAP<m>:DEFault <PrintColorSet>

Defines the default color set for printing of the display image.

To set the output to printer, use `HCOPy:DESTination<1..2>` with parameter `'SYST:COMM:PRIN'`.

Suffix:
<m> 1..2
Selects the output configuration.

Parameters:
<PrintColorSet> DEF1 | DEF4

DEF1
Current screen colors with white background and black grid.

DEF4
Current screen colors without any changes (black background).

*RST: DEF1

Usage: Asynchronous command

HCOPy:SSD <ShowSetupDialog>

Enables or disables the display of open dialog boxes in screenshots. Use this command if you want to document settings in screenshots.

Parameters:
<ShowSetupDialog> ON | OFF
*RST: OFF

Usage: Asynchronous command

Firmware/software: FW 3.20

HCOPy:ISBA IncludeSignalBar

If the value is set to `OFF`, the screenshot shows only the diagram area, without the signal bar.

Parameters:
IncludeSignalBar ON | OFF
*RST: OFF

Usage: Asynchronous command

HCOPy:IMMediate<m>[:DUM]

Starts the immediate output of the display image, depending on the `HCOPy:DESTination<1..2>` destination setting.

By default, screenshots are inverted when saving them with remote commands. If you want the screen display in the screenshot, turn inversion off (`HCOPy:DEVice<m>:INVerse`).

To get a correct screenshot of the diagrams, results, and dialog boxes, turn on the display using `SYST:DISP:UPD ON`.

Suffix:

<m> 1..2
Selects the output configuration.

Example:

```
SYST:DISP:UPD ON
HCOP:DEST 'MMEM'
MMEM:NAME 'C:\Temp\Print.bmp'
HCOP:IMMediate; *OPC?
```

Example: See Chapter 24.3.5.1, "Saving a screenshot to file", on page 1266

Usage: Event
Asynchronous command

HCOPy:IMMediate<m>:NEXT

Starts the output of the next display image, depending on the `HCOPy:DESTination<1..2>` destination setting.

If the screenshot is saved to a file, the file name used in the last saving process is automatically counted up to the next unused name.

Suffix:

<m> 1..2
Selects the output configuration.

Example: See Chapter 24.3.5.1, "Saving a screenshot to file", on page 1266

Usage: Event
Asynchronous command

24.16.9 Reports

The following commands configure and save report files. To configure the screenshot that is included in the report, use the commands explained in Chapter 24.16.8, "Screenshots", on page 1660.

REPort:LANGuage 1665
REPort:PAPersize 1665
REPort:LOGType 1665

REPort:LOGO....1665
REPort:USER....1666
REPort:COMMent....1666
REPort:FILE:NAME....1666
REPort:FILE:SAVE....1666

REPort:LANGuage <Language>

Sets the language to be used in the report. Available languages are listed in the data sheet.

Parameters:

<Language> String with the english language name, upper case.

Example: `REPort:LANGuage 'Spanish'`

REPort:PAPersize <PaperSize>

Selects the paper size: A4 or US Letter.

Parameters:

<PaperSize> A4 | USL

*RST: A4

Usage: Asynchronous command

REPort:LOGType <Logo>

By default, the Rohde & Schwarz logo is shown in the header of the report pages. You can switch the logo off, or select your logo to be shown.

Parameters:

<Logo> RS | CUST | NONE

CUST
Select the logo file using `REPort:LOGO`.

*RST: RS

Usage: Asynchronous command

REPort:LOGO <LogoFile>

Defines the logo file that is used on the report if `REPort:LOGType` is set to `CUSTom`.

Parameters:

<LogoFile> String with the path and filename of the logo image.

Example: `REPort:LOGO 'C:\Company files\logo.jpg'`

Usage: Asynchronous command

REPort:USER <User>

Enter the user name that appears in the general information section at the beginning of the report.

Parameters:
<User> String parameter

Usage: Asynchronous command

REPort:COMMent <Comment>

Enter a comment that appears in the general information section at the beginning of the report.

Parameters:
<Comment> String parameter

Usage: Asynchronous command

REPort:FILE:NAME <ReportFile>

Sets the file name and path to save the report to.

Parameters:
<ReportFile> String with path and file name. The file extension defines the file format: PDF, HTML, or DOC.

Usage: Asynchronous command

REPort:FILE:SAVE

Saves the report to the specified file.

Usage: Event
Asynchronous command

24.16.10 Autonaming

MMEMory:AUTonaming:PREFix 1666
MMEMory:AUTonaming:USERtext 1666
MMEMory:AUTonaming:DATE 1667
MMEMory:AUTonaming:INDex 1667
MMEMory:AUTonaming:TIME 1667
MMEMory:AUTonaming:TEXT 1667
MMEMory:AUTonaming:DEFaultpath 1667
MMEMory:AUTonaming:RESPath 1667
MMEMory:AUTonaming:RESall 1667

MMEMory:AUTonaming:PREFix <State>
MMEMory:AUTonaming:USERtext <State>

MMEMory:AUTonaming:DATE <State>
MMEMory:AUTonaming:INDex <State>
MMEMory:AUTonaming:TIME <State>

Includes or excludes the name part in the file name pattern for automatic file name generation. This name is used as the default file name.

The prefix indicates the type of data that is saved, for example, Histogram, RefCurve, Settings.

To define a user text, use `MMEMory:AUTonaming:TEXT`.

Parameters:
<State> ON | OFF
*RST: ON

Usage: Asynchronous command

MMEMory:AUTonaming:TEXT <NameString>

Defines a text, that can be included in the autonaming pattern.

Parameters:
<NameString> String parameter

Usage: Asynchronous command

MMEMory:AUTonaming:DEFaultpath <Path>

Sets the path where data and settings files will be stored. The factory default path is:

- `"C:\Users\Public\Documents\Rohde-Schwarz\RTx"` if no USB flash drive is connected
- Drive letter of the USB flash drive, for example, `"E:\"` or `"F:\"` if a USB flash drive is connected.

Parameters:
<Path> String parameter

Usage: Asynchronous command

MMEMory:AUTonaming:RESPath

Resets the path for file operations to the factory default path.

Usage: Event
Asynchronous command

MMEMory:AUTonaming:RESAll

Resets all autonaming settings to the default value, including the path.

Usage: Event
Asynchronous command

24.16.11 CSV export

EXPort:RESult:DECSymbol 1668
EXPort:RESult:DELimiter 1668

EXPort:RESult:DECSymbol <DecimalSymbol>

Sets if point or comma is used as a decimal symbol.

Parameters:
<DecimalSymbol> POINt | COMMa
*RST: POINt

Usage: Asynchronous command

EXPort:RESult:DELimiter <CsvDelimiter>

Selects the list separator symbol from a list.

Parameters:
<CsvDelimiter> SEMICOLON | COMMA | SPACE | TAB | COLON
*RST: COMMA

Usage: Asynchronous command

24.16.12 External application

The commands configure an external application that can be started at various events, for example, trigger event, or mask test violation.

EXECutable:NAME 1668
EXECutable:PARameter 1669
EXECutable:WDIRectory 1669

EXECutable:NAME <ApplicationPath>

Sets the path to the application executable.

Parameters:
<ApplicationPath> String parameter containing path, filename, and file extension

Example:
```
EXECutable:NAME 'C:\Program
Files\Wireshark\Wireshark.exe'
TRIGger:EVENt:RUNexec TRIGger
```

Usage: Asynchronous command

EXECutable:PARameter <AppParameters>

Sets optional parameters for the external executable.

Parameters:
<AppParameters> String parameter

Usage: Asynchronous command

EXECutable:WDIRectory <WorkDirectory>

Sets the working directory for the executable.

Parameters:
<WorkDirectory> String parameter

Usage: Asynchronous command

24.17 Protocols

- Configuration settings for all serial protocols 1670
- Trigger settings for all serial protocols 1674
- I²C (option R&S RTO6-K510) 1676
- SPI (option R&S RTO6-K510) 1708
- UART/RS-232/RS-422/RS-485 (option R&S RTO6-K510) 1727
- Audio signals (option R&S RTO6-K510) 1737
- Custom: manchester / NRZ (option R&S RTO6-K510) 1752
- CAN (option R&S RTO6-K520) 1795
- LIN (option R&S RTO6-K520) 1837
- Flexray (option R&S RTO6-K520) 1865
- SENT (option R&S RTO6-K520) 1899
- CXPI (option R&S RTO6-K520) 1929
- MIL-1553 (option R&S RTO6-K530) 1957
- ARINC 429 (option R&S RTO6-K530) 1982
- Space wire (option R&S RTO6-K530) 1999
- Ethernet 10BASE-T and 100BASE-TX (option R&S RTO6-K540) 2016
- MDIO (option R&S RTO6-K540) 2040
- RFFE (option R&S RTO6-K550) 2055
- Ethernet 100BASE-T1 (option R&S RTO6-K560) 2086
- Ethernet 1000BASE-T1 (option R&S RTO6-K560) 2116
- USB (option R&S RTO6-K570) 2147
- USB 3.1 (option R&S RTO6-K570) 2188
- USBPD (option R&S RTO6-K570) 2218
- D-PHY (Option R&S RTO6-K580) 2242
- M-PHY (option R&S RTO6-K580) 2274
- 8B/10B (option R&S RTO6-K590) 2331
- PCIe (option R&S RTO6K72) 2346
- DDR (option R&S RTO6-K91) 2387

24.17.1 Configuration settings for all serial protocols

BUS<m>:TYPE....1670
BUS<m>[:STATe]....1670
BUS<m>:SETReflevels....1671
BUS<m>:FAUToset....1671
BUS<m>:RESult....1671
BUS<m>:THReshold....1671
BUS<m>:FORMat....1672
BUS<m>:NEWList....1672
BUS<m>:SYMBols....1672
BUS<m>:ZCOupling....1672
DISPlay:RESultboxes:DEPosition....1673
BUS<m>:EXPResult:DETail....1673
BUS<m>:EXPResult:SAVE....1673
BUS<m>:EXPResult:TIME....1673

BUS<m>:TYPE <Type>

Defines the bus or protocol type for analysis. The type of available buses depends on the installed options.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Type> I2C | SPI | UART | CAN | CANFd | LIN | MILS1553 | MILStd1553 | ARIN429 | ARINc429 | SWIRe | MDIO | HBTO | USB | USBPD | ETHernet | CMSB | FLXRay | I2S | SENT | CXPI | RFFE | DPHY | MPHY | EBTB | USB3 | PCIE | TBTO

MILS1553 = MILStd1553: specification MIL-STD-1553
ARIN429 = ARINc429: specification ARINC 429
HBTO: Ethernet 100BASE-T1
CMSB: custom decode serial bus
SWIRe: SpaceWire

*RST: I2C

Usage: Asynchronous command

BUS<m>[:STATe] <State>

Enables the decoding of the specified bus.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

BUS<m>:SETReflevels

Executes the measurement of reference levels and sets the thresholds to the middle reference level of the measured amplitude.

Suffix:
<m> 1..4
Selects the serial bus.

Usage: Event
Asynchronous command

BUS<m>:FAUToset

Starts software algorithms for determining the signal threshold levels and bitrate.

Suffix:
<m> 1..4

Usage: Event
Asynchronous command

BUS<m>:RESult <ShowResultTable>

Opens a table with decoded data of the serial signal. The function affects all protocol types and requires the option for the analyzed protocol.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ShowResultTable> ON | OFF
*RST: ON

Usage: Asynchronous command

BUS<m>:THReshold <ShwThresLines>

If ON, the threshold levels are displayed in the diagram.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ShwThresLines> ON | OFF
*RST: OFF

Usage: Asynchronous command

BUS<m>:FORMat <DataFormat>

Sets the number format for decoded data values of the indicated serial bus. It defines the format in the decode table, and in the combs of the decoded signal on the screen.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<DataFormat> HEX | OCT | BIN | ASCII | ASCii | SIGN | USIG
ASCII = ASCii
USIG = unsigned
*RST: HEX

BUS<m>:NEWList <FileName>

Loads a label list file.

Suffix:
<m> 1..4
Selects the serial bus.

Setting parameters:
<FileName> String parameter with path and file name.

Example:
```
BUS1:NEWList 'C:\Protocols\CAN.csv'
BUS1:SYMBols ON
```

Usage: Setting only

BUS<m>:SYMBols <UseTranslation>

Activates the lable list to be used for decoding.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<UseTranslation> ON | OFF
*RST: OFF

Usage: Asynchronous command

BUS<m>:ZCOupling <ZoomCoupling>

If enabled, the protocol decode zoom and result table are synchronized.

Suffix:
<m> 1..4

Parameters:
<ZoomCoupling> ON | OFF
*RST: OFF

DISPlay:RESultboxes:DEPosition <DecodePosition>

Defines the position of the protocol decode result table on the screen.

Parameters:
<DecodePosition> PREV | FLOA | DOCK

PREV
Preview: result icon on the sidebar.

FLOA
Floating result box in front of the diagrams.

DOCK
Docked: fixed tab below the diagrams.

*RST: DOCK

Usage: Asynchronous command

BUS<m>:EXPResult:DETail <XptResDetails>

Includes the detailed results for all frames for the exported protocol results.

Suffix:
<m> 1..4

Parameters:
<XptResDetails> ON | OFF
*RST: ON

BUS<m>:EXPResult:SAVE <FileName>

Saves the selected results to the indicated file.

Suffix:
<m> 1..4

Setting parameters:
<FileName>

Usage: Setting only

BUS<m>:EXPResult:TIME <XptResTim>

Includes the frame timing for the exported protocol results.

Suffix:
<m> 1..4

Parameters:

<XptResTim> ON | OFF

*RST: ON

24.17.2 Trigger settings for all serial protocols

The following commands are available for all serial protocols that have a protocol trigger.

TRIGger<m>:SOURce[:SELect]....1674
TRIGger<m>:SOURce:SBSelect.... 1675
BUS<m>:TYPE....1675

TRIGger<m>:SOURce[:SELect] <SourceDetailed>

Selects the source of the trigger signal.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger
Available values depend on the selected trigger source. For input channels CHAN1...4, a trigger sequence can be configured.
For all other trigger sources, only suffix 1 is allowed.

Parameters:

<SourceDetailed> CHAN1 | CHANnel1 | CHAN2 | CHANnel2 | CHAN3 | CHANnel3 | CHAN4 | CHANnel4 | EXTernanalog | SBUS | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | LOGIC | MSOB1 | MSOB2 | MSOB3 | MSOB4 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4
Input channels

EXTernanalog
External analog signal connected to the External Trigger Input.
For this source, only the analog edge trigger is available.

SBUS
Serial bus

D0...D15
Digital channels (option R&S RTO6-B1)
See also: Chapter 24.18.4, "Trigger settings for digital signals and parallel buses", on page 2431

LOGIc
Logic combination of digital channels, used as trigger source (option R&S RTO6-B1)

MSOB1 | MSOB2 | MSOB3 | MSOB4
Parallel bus (option R&S RTO6-B1)

Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4
Input channels of R&S RT-ZVCmulti-channel power probe. Only available in the A-trigger with trigger type EDGE.

DIFF1 | DIFF2 | COMMON1 | COMMON2
Differential signals

*RST: CHAN1

Usage: Asynchronous command

TRIGger<m>:SOURce:SBSelect <SerialBus>

Selects the serial bus to be triggered on.

Suffix:
<m> 1..3
Event in a trigger sequence: 1 = A-event only

Parameters:
<SerialBus> SBUS1 | SBUS2 | SBUS3 | SBUS4

Usage: Asynchronous command

Firmware/software: Version 2.70

BUS<m>:TYPE <Type>

Defines the bus or protocol type for analysis. The type of available buses depends on the installed options.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Type> I2C | SPI | UART | CAN | CANFd | LIN | MILS1553 | MILStd1553 | ARIN429 | ARINc429 | SWIRe | MDIO | HBTO | USB | USBPD | ETHernet | CMSB | FLXRay | I2S | SENT | CXPI | RFFE | DPHY | MPHY | EBTB | USB3 | PCIE | TBTO

MILS1553 = MILStd1553: specification MIL-STD-1553
ARIN429 = ARINc429: specification ARINC 429
HBTO: Ethernet 100BASE-T1
CMSB: custom decode serial bus
SWIRe: SpaceWire

*RST: I2C

Usage: Asynchronous command

24.17.3 I²C (option R&S RTO6-K510)

- Configuration....1676
- Trigger....1678
- Decode results....1684
- I²C search settings....1693
- I²C search results....1701

24.17.3.1 Configuration

BUS<m>:I2C:SCL:SOURce....1676
BUS<m>:I2C:SDA:SOURce....1676
BUS<m>:I2C:SCL:THReshold....1677
BUS<m>:I2C:SDA:THReshold....1677
BUS<m>:I2C:TECHnology....1677
BUS<m>:I2C:RWBit....1678

BUS<m>:I2C:SCL:SOURce <SCLSource>

Sets the waveform of the clock line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<SCLSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for data and clock lines.
See Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C2W1

Usage: Asynchronous command

BUS<m>:I2C:SDA:SOURce <SDASource>

Sets the waveform of the data line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<SDASource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for data and clock lines.
See Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C1W1

Usage: Asynchronous command

BUS<m>:I2C:SCL:THReshold <SCLThreshold>

Sets a user-defined threshold value for the clock line.

Alternatively, you can set the threshold according to the signal technology with `BUS<m>:I2C:TECHnology`.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<SCLThreshold> User-defined clock threshold

Range: -12 to 12
Increment: 0.1
*RST: 0
Default unit: V

BUS<m>:I2C:SDA:THReshold <SDAThreshold>

Sets a user-defined threshold value for the data line.

Alternatively, you can set the threshold according to the signal technology with `BUS<m>:I2C:TECHnology`.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<SDAThreshold> User-defined data threshold

Range: -12 to 12
Increment: 0.1
*RST: 0
Default unit: V

BUS<m>:I2C:TECHnology <Technology>

Sets the threshold voltage clock and data lines as defined for various signal technologies.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Technology> V15 | V25 | V165 | V125 | V09 | VM13 | V38 | V20 | V0 | MANual

V15 | V25 | V165 | V125 | V09 | V38 | V20 | V0
1.5 V, 2.5 V, 1.65 V ... respectively

VM13
-1.3 V (negative value)

MANual
Manual setting of user-defined values with `BUS<m>:I2C:SCL:THReshold` and `BUS<m>:I2C:SDA:THReshold`.

*RST: V165

BUS<m>:I2C:RWBit <BusConfig>

Defines if the R/W bit of a 7-bit address is considered separately or as part of the address. 10-bit addresses are not affected. The setting defines which address lengths are available with `TRIGger<m>:I2C:AMODe`.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<BusConfig> SEParate | INADdress

SEParate
7-bit address and separate R/W bit.

INADdress
8-bit address with R/W bit included.

*RST: SEParate

24.17.3.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:I2C:TYPE....1679
TRIGger<m>:I2C:ACCess....1679
TRIGger<m>:I2C:ADNack....1680
TRIGger<m>:I2C:DWNack....1680
TRIGger<m>:I2C:DRNack....1680

TRIGger<m>:I2C:AMODe 1680
TRIGger<m>:I2C:ACONdition 1681
TRIGger<m>:I2C:ADDRess 1681
TRIGger<m>:I2C:ADDTo 1681
TRIGger<m>:I2C:ADOR<n>:ENABle 1681
TRIGger<m>:I2C:ADOR<n>:ADRType 1682
TRIGger<m>:I2C:ADOR<n>[:VALue] 1682
TRIGger<m>:I2C:ADOR<n>:RWBit 1682
TRIGger<m>:I2C:DPOPerator 1683
TRIGger<m>:I2C:DPOSition 1683
TRIGger<m>:I2C:DPTO 1683
TRIGger<m>:I2C:DCONdition 1683
TRIGger<m>:I2C:DMIN 1684
TRIGger<m>:I2C:DMAX 1684

TRIGger<m>:I2C:TYPE <Type>

Selects the trigger type for I²C analysis.

Parameters:

<Type> STARt | REPStart | STOP | NACK | ADDRess | ADOR | ADAT

STARt
Start condition

REPStart
Repeated start - the start condition occurs without previous stop condition.

NACK
Missing acknowledge bit. To localize specific missing acknowledge bits, use `TRIGger<m>:I2C:ADNack`, `TRIGger<m>:I2C:DWNack`, and `TRIGger<m>:I2C:DRNack`.

ADDRess
Triggers on one specific address

ADOR
Triggers on an OR combination with up to four four address conditions.

ADAT
Triggers on a combination of address and data condition.

STOP
Stop condition, end of frame

*RST: STARt

Usage: Asynchronous command

TRIGger<m>:I2C:ACCess <RWBitAddress>

Sets the trigger condition for the R/W bit - the transfer direction of the data.

Parameters:

<RWBitAddress> READ | WRITe | EITHer

EITHer
Transfer direction is not relevant.

*RST: EITHer

Usage: Asynchronous command

TRIGger<m>:I2C:ADNack <AddressNack>

Triggers if the address acknowledge bit is missing - no slave recognizes the address.

Parameters:

<AddressNack> ON | OFF

*RST: ON

Usage: Asynchronous command

TRIGger<m>:I2C:DWNack <DataWriteNack>

Triggers if a date acknowledge bit is missing - the addressed slave does not accept the data.

Parameters:

<DataWriteNack> ON | OFF

*RST: ON

Usage: Asynchronous command

TRIGger<m>:I2C:DRNack <DataReadNack>

Triggers on the end of the read process when the master reads data from the slave. This Nack is sent according to the protocol definition, it is not an error.

Parameters:

<DataReadNack> ON | OFF

*RST: ON

Usage: Asynchronous command

TRIGger<m>:I2C:AMODe <AddressType>

Sets the address length. The setting affects the address input with `TRIGger<m>:I2C:ADDRess` and `TRIGger<m>:I2C:ADDTo`.

Parameters:

<AddressType> BIT7 | BIT7_RW | BIT10 | ANY

BIT7
Enter the 7 address bits. Only available if `BUS<m>:I2C:RWBitSEParate` is set.

BIT7_RW
Enter 7 address bits and the R/W bit. Only available if `BUS<m>:I2C:RWBitINADdress` is set.

BIT10
10-bit address

ANY
Only available for trigger type "Address + data" (`TRIGger<m>:I2C:TYPE ADAT`). Used to trigger on data only, regardless of the address.

*RST: BIT7

Usage: Asynchronous command

TRIGger<m>:I2C:ACONdition <AddressOperator>

Sets the operator to set a specific address or an address range. The address values are set with `TRIGger<m>:I2C:ADDRess` and `TRIGger<m>:I2C:ADDTo`.

Parameters:
<AddressOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

*RST: EQUal

TRIGger<m>:I2C:ADDRess <Address>

Triggers on the specified slave address, or sets the the start value of an address range depending on the condition set with `TRIGger<m>:I2C:ACONdition`.

Parameters:
<Address> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

TRIGger<m>:I2C:ADDTo <AddressTo>

Sets the the end value of an address range if the condition is set to an address range with `TRIGger<m>:I2C:ACONdition`.

Parameters:
<AddressTo> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

Usage: Asynchronous command

TRIGger<m>:I2C:ADOR<n>:ENABle <UseAddress>

Includes the indicated ADOR address in the "address OR" trigger condition.

Suffix:
<n> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:

<UseAddress> ON | OFF

*RST: OFF

TRIGger<m>:I2C:ADOR<n>:ADRType <AddressType>

Sets the address type for the indicated ADOR address in the "address OR" trigger condition.

Suffix:

<n> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:

<AddressType> BIT7 | BIT7_RW | BIT10

*RST: BIT7

TRIGger<m>:I2C:ADOR<n>[:VALue] <Address>

Defines the address pattern of the indicated ADOR address in the "address OR" trigger condition.

Suffix:

<n> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:

<Address> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

TRIGger<m>:I2C:ADOR<n>:RWBit <RWBit>

Defines the R/W bit of the indicated ADOR address in the "address OR" trigger condition.

Suffix:

<n> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:

<RWBit> UNDefined | READ | WRITe | EITHer

UNDefined
Return value only

*RST: EITHer

TRIGger<m>:I2C:DPOPerator <DataPosOperator>

Sets the operator for the data position. You can defined an exact position, or a position range.

Parameters:

<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe

ANY = OFF
The position of the required pattern is not relevant.

EQUal | GETHan
Equal, Greater or equal than. These conditions require one data position to be set with `TRIGger<m>:I2C:DPOSition`.

INRange = RANGe
In range: Set the minimum and maximum value of the range with `TRIGger<m>:I2C:DPOSition` and `TRIGger<m>:I2C:DPTO`.

*RST: ANY

TRIGger<m>:I2C:DPOSition <DataPosition>

Sets the number of data bytes before the first byte of interest. These bytes are ignored.

Parameters:

<DataPosition> The index 0 is associated with the first data byte.

Range: 0 to 4095
Increment: 1
*RST: 0

TRIGger<m>:I2C:DPTO <DataPositionTo>

Defines the last byte of interest, if `TRIGger<m>:I2C:DPOPerator` is set to `RANGe`.

Parameters:

<DataPositionTo> Range: 0 to 4095
Increment: 1
*RST: 0

TRIGger<m>:I2C:DCONdition <DataOperator>

Sets the operator to set a specific data value or a data range.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

*RST: EQUal

TRIGger<m>:I2C:DMIN <Data>

Specifies the data bit pattern, or sets the the start value of a data pattern range.. Enter the bytes in msb first bit order. The maximum pattern length is 64 bit. Waveform data is compared with the pattern byte-by-byte.

Parameters:

<Data> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Usage: Asynchronous command

TRIGger<m>:I2C:DMAX <DataTo>

Sets the the end value of an data range if `TRIGger<m>:I2C:DCONdition` is set to `INRange` or `OORange`.

Parameters:

<DataTo> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Usage: Asynchronous command

24.17.3.3 Decode results

To load and acivate a label list, use:

- `BUS<m>:NEWList` on page 1672
- `BUS<m>:SYMBols` on page 1672

BUS<m>:I2C:FRAMe<n>:DATA?....1685
BUS<m>:I2C:FCOunt?....1685
BUS<m>:I2C:FRAMe<n>:AACCess?....1685
BUS<m>:I2C:FRAMe<n>:ACCess?....1686
BUS<m>:I2C:FRAMe<n>:ACOMplete?....1686
BUS<m>:I2C:FRAMe<n>:ADBStart?....1686
BUS<m>:I2C:FRAMe<n>:ADDRess?....1687
BUS<m>:I2C:FRAMe<n>:ADEVice?....1687
BUS<m>:I2C:FRAMe<n>:AMODe?....1687
BUS<m>:I2C:FRAMe<n>:ASTart?....1688
BUS<m>:I2C:FRAMe<n>:BITRate?....1688
BUS<m>:I2C:FRAMe<n>:RWBStart?....1688
BUS<m>:I2C:FRAMe<n>:STATus?....1689
BUS<m>:I2C:FRAMe<n>:STARt?....1689
BUS<m>:I2C:FRAMe<n>:STOP?....1690
BUS<m>:I2C:FRAMe<n>:SYMBol?....1690
BUS<m>:I2C:FRAMe<n>:BCOunt?....1690
BUS<m>:I2C:FRAMe<n>:BYTE<o>:ACCess?....1691
BUS<m>:I2C:FRAMe<n>:BYTE<o>:ACKStart?....1691

BUS<m>:I2C:FRAMe<n>:BYTE<o>:COMPlete?.....1691
BUS<m>:I2C:FRAMe<n>:BYTE<o>:STARt?.....1692
BUS<m>:I2C:FRAMe<n>:BYTE<o>:VALue?.....1692

BUS<m>:I2C:FRAMe<n>:DATA?

Returns the data words of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Data> Comma-separated list of integer values (N, D1, D2,…, DN). N is the number of bytes in the frame, and D1…DN are the values of the bytes.

Example:

```
BUS:I2C:FRAMe4:DATA?
<-- 3,74,164,18
```

Usage: Query only

BUS<m>:I2C:FCOunt?

Returns the number of decoded frames.

Suffix:

<m> 1..4
Selects the serial bus.

Return values:

<Count> Total number of decoded frames.

Usage: Query only

BUS<m>:I2C:FRAMe<n>:AACCess?

Returns the address acknowledge bit value for the indicated frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<AddressAckBit> INComplete | ACK | NACK | EITHer
*RST: INComplete

Usage: Query only

BUS<m>:I2C:FRAMe<n>:ACCess?

Returns the value of the R/W bit of the indicated frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<RWBit> UNDefined | READ | WRITe | EITHer
*RST: UNDefined

Usage: Query only

BUS<m>:I2C:FRAMe<n>:ACOMplete?

Returns if the address is completely contained in the acquisition.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<AddressComplete> ON | OFF
*RST: OFF

Usage: Query only

BUS<m>:I2C:FRAMe<n>:ADBStart?

Returns the start time of the address acknowledge bit.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<AddrAckBtStrt> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:I2C:FRAMe<n>:ADDRess?

Returns the device address value of the indicated frame, that is, the address value that is shown in the decoded cells and in the decode results table.

If the frame has a 7-bit address, the command considers the status of `BUS<m>:I2C:RWBit`. If `BUS<m>:I2C:RWBit INADdress` is set, the returned address includes the R/W bit (8 bit). Otherwise, the pure address without the R/W bit is returned (7 bit, same result as returned with `BUS<m>:I2C:FRAMe<n>:ADEVice?`).

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<AddressValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and default – are decimal values.

Range: 0 to 1023
*RST: 0

Usage: Query only

BUS<m>:I2C:FRAMe<n>:ADEVice?

Returns the pure device address of the indicated frame *without* the R/W bit.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<DeviceAddress> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and default – are decimal values.

Range: 0 to 1023
*RST: 0

Usage: Query only

BUS<m>:I2C:FRAMe<n>:AMODe?

Returns the address length.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<AddressType> BIT7 | BIT7_RW | BIT10 | AUTO | ANY
*RST: BIT7

Usage: Query only

BUS<m>:I2C:FRAMe<n>:ASTart?

Returns the start time of the address for the indicated frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<AddressStart> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:I2C:FRAMe<n>:BITRate?

Returns the primary bit rate.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<PrimaryBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

BUS<m>:I2C:FRAMe<n>:RWBStart?

Returns the start time of the R/W bit

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<RWBitStart> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:I2C:FRAMe<n>:STATus?

Returns the overall state of the frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<FrameState> INComplete | OK | UNEXpstop | INSufficient | ADDifferent

INComplete
The stop bit is missing.

OK
The frame is valid.

UNEXpstop
A stop bit was detected but clock and data are continued.

INSufficient
The frame is not completely contained in the acquisition. The acquired part of the frame is valid.

ADDifferent
Error in 10 bit address. In case of a read access on a 10 bit address, the first address byte is sent twice, first as write, the second as read. The first seven bits of the byte must be identical. If they are not identical, the ADDifferent error is indicated.

*RST: OK

Usage: Query only

BUS<m>:I2C:FRAMe<n>:STARt?

Returns the start time of the specified frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameStart> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:I2C:FRAMe<n>:STOP?

Returns the end time of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameStop> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:I2C:FRAMe<n>:SYMBol?

Returns the symbolic label of the specified frame if the label list is enabled.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the number of the frame in the current acquisition, 1...n.

Return values:

<Translation> String with symbolic name of the address

Usage: Query only

BUS<m>:I2C:FRAMe<n>:BCOunt?

Returns the number of bytes in the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Count> Byte count

Usage: Query only

BUS<m>:I2C:FRAMe<n>:BYTE<o>:ACCess?

Returns the acknowledge bit value of the specified data byte.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte number.

Return values:

<AckBit> INComplete | ACK | NACK | EITHer

*RST: INComplete

Usage: Query only

BUS<m>:I2C:FRAMe<n>:BYTE<o>:ACKStart?

Returns the start time of the acknowledge bit of the specified byte.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte number.

Return values:

<AckBitStart> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:I2C:FRAMe<n>:BYTE<o>:COMPlete?

Returns if the indicated byte is completely contained in the acquisition.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte number.

Return values:
<ValueComplete> ON | OFF
*RST: OFF

Usage: Query only

BUS<m>:I2C:FRAMe<n>:BYTE<o>:STARt?

Returns the start time of the specified data byte.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte number.

Return values:
<Start> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:I2C:FRAMe<n>:BYTE<o>:VALue?

Returns the data value of the specified byte.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte number.

Return values:
<Value> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and default – are decimal values.
Range: 0 to 255
*RST: 0

Usage: Query only

24.17.3.4 I²C search settings

In search setup commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name. The commands are similar to I²C trigger commands.

SEARch:TRIGger:I2C:SCONdition....1693
SEARch:TRIGger:I2C:RCONdition....1693
SEARch:TRIGger:I2C:STCNdition....1694
SEARch:TRIGger:I2C:NACKnowledge....1694
SEARch:TRIGger:I2C:SADDress....1694
SEARch:TRIGger:I2C:ADOR....1695
SEARch:TRIGger:I2C:ADData....1695
SEARch:TRIGger:I2C:ACONdition....1695
SEARch:TRIGger:I2C:AMODe....1696
SEARch:TRIGger:I2C:ADDRess....1696
SEARch:TRIGger:I2C:ADDTo....1696
SEARch:TRIGger:I2C:ACCess....1697
SEARch:TRIGger:I2C:ADDO<m>:ENABle....1697
SEARch:TRIGger:I2C:ADDO<m>:ADRType....1697
SEARch:TRIGger:I2C:ADDO<m>[:VALue]....1698
SEARch:TRIGger:I2C:ADDO<m>:RWBit....1698
SEARch:TRIGger:I2C:DPOPerator....1698
SEARch:TRIGger:I2C:DPOSition....1699
SEARch:TRIGger:I2C:DPTO....1699
SEARch:TRIGger:I2C:DCONdition....1699
SEARch:TRIGger:I2C:DMIN....1700
SEARch:TRIGger:I2C:DMAX....1700
SEARch:TRIGger:I2C:ADNack....1700
SEARch:TRIGger:I2C:DRNack....1700
SEARch:TRIGger:I2C:DWNack....1701

SEARch:TRIGger:I2C:SCONdition <SearchName>,<Start>
SEARch:TRIGger:I2C:SCONdition? <SearchName>

Enables the search for the start of the message.

Parameters:
<Start> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:RCONdition <SearchName>,<RepeatedStart>
SEARch:TRIGger:I2C:RCONdition? <SearchName>

Enables the search for a start condition without previous stop condition.

Parameters:
<RepeatedStart> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:STCNdition <SearchName>,<Stop>
SEARch:TRIGger:I2C:STCNdition? <SearchName>

Enables the search for the start of the message.

Parameters:
<Stop> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:NACKnowledge <SearchName>,<NoAcknowledge>
SEARch:TRIGger:I2C:NACKnowledge? <SearchName>

Searches for missing address acknowledge bits.

Parameters:
<NoAcknowledge> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:SADDress <SearchName>,<Address>
SEARch:TRIGger:I2C:SADDress? <SearchName>

Enables the search for one specific address condition or for a combination of address conditions.

To define the address condition, use the following commands:

- `SEARch:TRIGger:I2C:ACONdition` on page 1695
- `SEARch:TRIGger:I2C:ADDRess` on page 1696
- `SEARch:TRIGger:I2C:ADDTo` on page 1696
- `SEARch:TRIGger:I2C:AMODe` on page 1696
- `SEARch:TRIGger:I2C:ACCess` on page 1697

Parameters:
<Address> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:ADOR <SearchName>,<AddressOr>
SEARch:TRIGger:I2C:ADOR? <SearchName>

Enables the search for one to four address conditions.

- `SEARch:TRIGger:I2C:ADDO<m>:ENABle` on page 1697
- `SEARch:TRIGger:I2C:ADDO<m>:ADRType` on page 1697
- `SEARch:TRIGger:I2C:ADDO<m>[:VALue]` on page 1698
- `SEARch:TRIGger:I2C:ADDO<m>:RWBit` on page 1698

Parameters:
<AddressOr> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:ADData <SearchName>,<AddressData>
SEARch:TRIGger:I2C:ADData? <SearchName>

Enables the search for a combination of address and data conditions.

Parameters:
<AddressData> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:ACONdition <SearchName>,<AddressOperator>
SEARch:TRIGger:I2C:ACONdition? <SearchName>

Sets the operator to set a specific address or an address range.

Parameters:
<AddressOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These condition require one identifier pattern to be set with `SEARch:TRIGger:I2C:ADDRess` on page 1696.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `SEARch:TRIGger:I2C:ADDRess` on page 1696 and `SEARch:TRIGger:I2C:ADDTo` on page 1696.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:AMODe <SearchName>,<AddressType>
SEARch:TRIGger:I2C:AMODe? <SearchName>

Sets the address length.

Parameters:

<AddressType> BIT7 | BIT7_RW | BIT10 | ANY

BIT7 | BIT10
Enter only address bits in the address pattern.

BIT7_RW
Enter seven address bits and also the R/W bit in the address pattern.

ANY
Only available for search criteria "Address and data" (`SEARch:TRIGger:I2C:ADData` is set `ON`). Used to search for data only, regardless of the address.

*RST: BIT7

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:I2C:ADDRess <SearchName>,<Address>
SEARch:TRIGger:I2C:ADDRess? <SearchName>

Specifies an address pattern, or sets the the start value of an address range.

Parameters:

<Address> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.
The pattern length is defined with `SEARch:TRIGger:I2C:AMODe`.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:ADDTo <SearchName>,<AddressTo>
SEARch:TRIGger:I2C:ADDTo? <SearchName>

Sets the the end value of an address range if `SEARch:TRIGger:I2C:ACONdition` is set to `INRange` or `OORange`.

Parameters:

<AddressTo> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.
The pattern length is defined with `SEARch:TRIGger:I2C:AMODe`.

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:I2C:ACCess <SearchName>,<RWBitAddress>
SEARch:TRIGger:I2C:ACCess? <SearchName>

Sets the transfer direction of the data.

Parameters:
<RWBitAddress> READ | WRITe | EITHer
*RST: EITHer

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:I2C:ADDO<m>:ENABle <SearchName>,<UseAddress>
SEARch:TRIGger:I2C:ADDO<m>:ENABle? <SearchName>

Includes the indicated ADOR address in the "address OR" search condition.

Suffix:
<m> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:
<UseAddress> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:ADDO<m>:ADRType <SearchName>,<AddressType>
SEARch:TRIGger:I2C:ADDO<m>:ADRType? <SearchName>

Sets the address type for the indicated ADOR address in the "address OR" search condition.

Suffix:
<m> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:
<AddressType> BIT7 | BIT7_RW | BIT10

BIT7 | BIT10
Enter only address bits in the address pattern.

BIT7_RW
Enter seven address bits and also the R/W bit in the address pattern.

*RST: BIT7

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:ADDO<m>[:VALue] <SearchName>,<Address>
SEARch:TRIGger:I2C:ADDO<m>[:VALue]? <SearchName>

Defines the address pattern of the indicated ADOR address in the "address OR" search condition.

Suffix:
<m> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:
<Address> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.
The pattern length is defined with `SEARch:TRIGger:I2C:ADDO<m>:ADRType`.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:ADDO<m>:RWBit <SearchName>,<RWBit>
SEARch:TRIGger:I2C:ADDO<m>:RWBit? <SearchName>

Defines the R/W bit of the indicated ADOR address in the "address OR" search condition.

Suffix:
<m> 1..4
Index of the address in an "address OR" condition (OR slot)

Parameters:
<RWBit> UNDefined | READ | WRITe | EITHer

UNDefined
Only return value

*RST: EITHer

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:DPOPerator <SearchName>,<DataPosOperator>
SEARch:TRIGger:I2C:DPOPerator? <SearchName>

Sets the operator for the data position. You can defined an exact position, or a position range.

Parameters:
<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe

ANY = OFF
The position of the required pattern is not relevant.

EQUal | GETHan
Equal, Greater or equal than. These conditions require one data position to be set with `SEARch:TRIGger:I2C:DPOSition`.

INRange = RANGe
In range: Set the minimum and maximum value of the range with `SEARch:TRIGger:I2C:DPOSition` and `SEARch:TRIGger:I2C:DPTO`.

*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:DPOSition <SearchName>,<DataPosition>
SEARch:TRIGger:I2C:DPOSition? <SearchName>

Defines the first byte of interest. All bytes before that byte are ignored.

Parameters:
<DataPosition> The index 0 is associated with the first data byte.

Range: 0 to 4095
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:DPTO <SearchName>,<DataPositionTo>
SEARch:TRIGger:I2C:DPTO? <SearchName>

Defines the last byte of interest, if `SEARch:TRIGger:I2C:DPOPerator` defines a range.

Parameters:
<DataPositionTo> Range: 0 to 4095
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:I2C:DCONdition? <SearchName>

Sets the operator to set a specific data value or a data range.

Parameters:
<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:I2C:DMIN <SearchName>,<Data>
SEARch:TRIGger:I2C:DMIN? <SearchName>

Specifies the data bit pattern, or sets the the start value of a data pattern range.. Enter the bytes in msb first bit order. The maximum pattern length is 64 bit. Waveform data is compared with the pattern byte-by-byte.

Parameters:
<Data> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:I2C:DMAX <SearchName>,<DataTo>
SEARch:TRIGger:I2C:DMAX? <SearchName>

Sets the the end value of an address range if `SEARch:TRIGger:I2C:DCONdition` is set to `INRange` or `OORange`.

Parameters:
<DataTo> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:I2C:ADNack <SearchName>,<AddressNack>
SEARch:TRIGger:I2C:ADNack? <SearchName>

Parameters:
<AddressNack> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:I2C:DRNack <SearchName>,<DataReadNack>
SEARch:TRIGger:I2C:DRNack? <SearchName>

Searches for the end of the read process when the master reads data from the slave. This Nack is sent according to the protocol definition, it is not an error.

Parameters:
<DataReadNack> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:I2C:DWNack <SearchName>,<DataWriteNack>
SEARch:TRIGger:I2C:DWNack? <SearchName>

Searches for missing data write acknowledge bits.

Parameters:
<DataWriteNack> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

24.17.3.5 I²C search results

The search on decoded CAN data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 24.17.3.3, "Decode results", on page 1684.

SEARch:RESult:I2C:FCOunt?.....1702
SEARch:RESult:I2C:FRAMe<m>:STATus?.....1702
SEARch:RESult:I2C:FRAMe<m>:STARt?.....1702
SEARch:RESult:I2C:FRAMe<m>:STOP?.....1702
SEARch:RESult:I2C:FRAMe<m>:AACCess?.....1703
SEARch:RESult:I2C:FRAMe<m>:ACCess?.....1703
SEARch:RESult:I2C:FRAMe<m>:ACOMplete?.....1703
SEARch:RESult:I2C:FRAMe<m>:ADBStart?.....1704
SEARch:RESult:I2C:FRAMe<m>:ADDRess?.....1704
SEARch:RESult:I2C:FRAMe<m>:ADEVice?.....1704
SEARch:RESult:I2C:FRAMe<m>:AMODe?.....1704
SEARch:RESult:I2C:FRAMe<m>:ASTart?.....1705
SEARch:RESult:I2C:FRAMe<m>:DATA?.....1705
SEARch:RESult:I2C:FRAMe<m>:RWBStart?.....1705
SEARch:RESult:I2C:FRAMe<m>:SYMBol?.....1706
SEARch:RESult:I2C:FRAMe<m>:BCOunt?.....1706
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:ACCess?.....1706
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:ACKStart?.....1706

SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:COMPlete?..... 1707
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:STARt?.....1707
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:VALue?.....1707

SEARch:RESult:I2C:FCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:
<Count> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:STATus? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> INComplete | OK | UNEXpstop | INSufficient | ADDifferent
*RST: OK

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:STOP? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:AACCess? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<AddressAckBit> INComplete | ACK | NACK | EITHer
*RST: INComplete

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:ACCess? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<RWBit> UNDefined | READ | WRITe | EITHer
*RST: UNDefined

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:ACOMplete? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<AddressComplete> ON | OFF
*RST: OFF

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:ADBStart? <SearchName>

Returns the start time of the address acknowledge bit.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<AddrAckBtStrt> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:ADDRess? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<AddressValue> Range: 0 to 2047
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:ADEVice? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<DeviceAddress> Range: 0 to 1023
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:AMODe? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<AddressType> BIT7 | BIT7_RW | BIT10 | AUTO | ANY
*RST: BIT7

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:ASTart? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<AddressStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:DATA? <SearchName>

Returns the data bytes of the indicated frame.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:RWBStart? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<RWBitStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:SYMBol? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Translation>

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:BCOunt?

Suffix:
<m> *

Return values:
<Count>

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:ACCess? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<ByteAckBit> INComplete | ACK | NACK | EITHer

*RST: INComplete

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:ACKStart? <SearchName>

Returns the start time of the acknowledge bit of the indicated data byte.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:

<ByteAckBitStart>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:COMPlete? <SearchName>

Suffix:

<m>	*
<n>	*

Query parameters:
<SearchName>

Return values:

<ByteComplete>	ON | OFF	
	*RST:	OFF

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:STARt? <SearchName>

Suffix:

<m>	*
<n>	*

Query parameters:
<SearchName>

Return values:

<ByteStart>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:VALue? <SearchName>

Suffix:

<m>	*
<n>	*

Query parameters:
<SearchName>

Return values:

<Value> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

24.17.4 SPI (option R&S RTO6-K510)

- SPI bus configuration....1708
- SPI trigger....1712
- SPI decode results....1715
- SPI search settings....1720
- SPI search results....1723

24.17.4.1 SPI bus configuration

BUS<m>:SPI:BORDer....1708
BUS<m>:SPI:WSIZe....1709
BUS<m>:SPI:SCLK:SOURce....1709
BUS<m>:SPI:SSELect:SOURce....1709
BUS<m>:SPI:SSELect:POLarity....1710
BUS<m>:SPI:MISO:SOURce....1710
BUS<m>:SPI:MISO:POLarity....1710
BUS<m>:SPI:MOSI:SOURce....1711
BUS<m>:SPI:MOSI:POLarity....1711
BUS<m>:SPI:SCLK:THReshold....1711
BUS<m>:SPI:MISO:THReshold....1711
BUS<m>:SPI:MOSI:THReshold....1711
BUS<m>:SPI:SSELect:THReshold....1711
BUS<m>:SPI:FRCondition....1712
BUS<m>:SPI:TIMeout....1712

BUS<m>:SPI:BORDer <BitOrder>

Defines if the data of the messages starts with msb (most significant bit) or lsb (least significant bit).

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<BitOrder> LSBF | MSBF
*RST: MSBF

Usage: Asynchronous command

BUS<m>:SPI:WSIZe <WordLength>

Sets the number of bits in a message.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<WordLength> Number of bits

Range: 4 to 32
Increment: 1
*RST: 8

Usage: Asynchronous command

BUS<m>:SPI:SCLK:SOURce <SCLKSource>

Sets the input channel of the clock line.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<SCLKSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for data, clock and slave select lines.
See Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C2W1

Usage: Asynchronous command

BUS<m>:SPI:SSELect:SOURce <SlaveSelectSource>

Sets the input channel of the slave select line.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<SlaveSelectSource> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for data, clock and slave select lines.
See Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: None

Usage: Asynchronous command

BUS<m>:SPI:SSELect:POLarity <SSPolarity>

Selects whether transmitted slave select signal is high active (high = 1) or low active (low = 1).

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<SSPolarity> ACTLow | ACTHigh
*RST: ACTLow

Usage: Asynchronous command

BUS<m>:SPI:MISO:SOURce <MISOSource>

Sets the input channel of the MISO line.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<MISOSource> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15
Digital and analog channels cannot be used at the same time for data, clock and slave select lines.
See Chapter 24.4.2, "Waveform parameter", on page 1284
*RST: None

Usage: Asynchronous command

BUS<m>:SPI:MISO:POLarity <MISOPolarity>

Selects whether transmitted data is high active (high = 1) or low active (low = 1).

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<MISOPolarity> ACTLow | ACTHigh
*RST: ACTHigh

Usage: Asynchronous command

BUS<m>:SPI:MOSI:SOURce <MOSISource>

Sets the input channel of the MOSI line.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<MOSISource> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for data, clock and slave select lines.
See Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C1W1

Usage: Asynchronous command

BUS<m>:SPI:MOSI:POLarity <MOSIPolarity>

Selects whether transmitted data is high active (high = 1) or low active (low = 1).

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<MOSIPolarity> ACTLow | ACTHigh

*RST: ACTHigh

Usage: Asynchronous command

BUS<m>:SPI:SCLK:THReshold <SCLKThreshold>
BUS<m>:SPI:MISO:THReshold <MISOThreshold>
BUS<m>:SPI:MOSI:THReshold <MOSIThreshold>
BUS<m>:SPI:SSELect:THReshold <SSThreshold>

Set user-defined threshold values for the clock, MISO, MOSI and slave select lines.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<SSThreshold> User-defined value

Range: -12 to 12
Increment: 0.1
*RST: 0
Default unit: V

BUS<m>:SPI:FRCondition <FrameCondition>

Defines the start of a frame. A frame contains a number of successive words, at least one word.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<FrameCondition> SS | CLKTimeout

SS
Start and end of the frame is defined by the active state of the slave select signal, see `BUS<m>:SPI:SSELect:POLarity`.

CLKTimeout
Defines a timeout on the clock line SCLK as limiter between two frames. The timeout condition is used for SPI connections without an SS line.

*RST: SS

BUS<m>:SPI:TIMeout <ClockTimeout>

Defines a timeout on the clock line SCLK as limiter between two frames. The timeout condition is used for SPI connections without an SS line.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ClockTimeout> Range: 50E-9 to 10
Increment: 1E-6
*RST: 1E-3
Default unit: s

24.17.4.2 SPI trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:SPI:TYPE....1713
TRIGger<m>:SPI:PALignment....1713
TRIGger<m>:SPI:DPOPerator....1714
TRIGger<m>:SPI:DPOSition....1714

TRIGger<m>:SPI:DPTO 1714
TRIGger<m>:SPI:FCONdition 1714
TRIGger<m>:SPI:MISopattern 1715
TRIGger<m>:SPI:MOSipattern 1715

TRIGger<m>:SPI:TYPE <Type>

Selects the trigger type for SPI analysis.

Parameters:

<Type> SSACtive | TIMeout | MOSI | MISO | MOMI

SSACtive
Start of the message: slave select signal SS changes to the active state.

TIMeout
Triggers on the next message start after the "Timeout" time.

MOSI
Triggers on a specified data pattern in that is expected on the MOSI line. Define the pattern with `TRIGger<m>:SPI:MOSipattern`.

MISO
Triggers on a specified data pattern in that is expected on the MISO line. Define the pattern with `TRIGger<m>:SPI:MISopattern`

MOMI
Triggers on a specified data patterns on the MISO and MISO lines.

*RST: SSACtive

Usage: Asynchronous command

TRIGger<m>:SPI:PALignment <DataAlignment>

Defines how the specified data pattern is searched.

Parameters:

<DataAlignment> WORD | BIT

WORD
The pattern is matched only at word boundaries.

BIT
Bit-by bit: the pattern can be at any position in the data word.

*RST: WORD

Usage: Asynchronous command

TRIGger<m>:SPI:DPOPerator <DataPosOperator>

Sets the operator for the data position. You can defined an exact position, or a position range.

Parameters:

<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe

ANY = OFF
The position of the required pattern is not relevant.

EQUal | GETHan
Equal, Greater or equal than. These conditions require one data position to be set with `TRIGger<m>:SPI:DPOSition`.

INRange = RANGe
Set the minimum and maximum value of the range with `TRIGger<m>:SPI:DPOSition` and `TRIGger<m>:SPI:DPTO`.

*RST: ANY

TRIGger<m>:SPI:DPOSition <DataPosition>

Sets the number of bits or words to be ignored before the first bit or word od interest. The effect is defined by `TRIGger<m>:SPI:PALignment`.

Parameters:

<DataPosition> The index 0 is associated with the first data byte.

Range: 0 to 4095 for triggering on one line (MISO or MOSI), 2047 for triggering on both lines.
Increment: 1
*RST: 0

TRIGger<m>:SPI:DPTO <DataPositionTo>

Defines the last bit or word of interest, if `TRIGger<m>:SPI:DPOPerator` is set to `INRange`.

Parameters:

<DataPositionTo> Range: 1 to 4095 for triggering on one line (MISO or MOSI), 2047 for triggering on both lines.
Increment: 1
*RST: 1

TRIGger<m>:SPI:FCONdition <DataOperator>

Selects the operator for the MISO and MOSI pattern.

Parameters:

<DataOperator> EQUal | NEQual

*RST: EQUal

TRIGger<m>:SPI:MISopattern <MISOPattern>

Specifies the pattern to be triggered on the MISO line.

Parameters:

<MISOPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Usage: Asynchronous command

TRIGger<m>:SPI:MOSipattern <MOSIPattern>

Specifies the pattern to be triggered on the MOSI line.

Parameters:

<MOSIPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Usage: Asynchronous command

24.17.4.3 SPI decode results

BUS<m>:SPI:FRAMe<n>:DATA?....1715
BUS<m>:SPI:FCOunt?....1716
BUS<m>:SPI:FRAMe<n>:COUNt?....1716
BUS<m>:SPI:FRAMe<n>:BITRate?....1716
BUS<m>:SPI:FRAMe<n>:STATus?....1716
BUS<m>:SPI:FRAMe<n>:STARt?....1717
BUS<m>:SPI:FRAMe<n>:STOP?....1717
BUS<m>:SPI:FRAMe<n>:WCOunt?....1718
BUS<m>:SPI:FRAMe<n>:WORD<o>:STARt?....1718
BUS<m>:SPI:FRAMe<n>:WORD<o>:STOP?....1718
BUS<m>:SPI:FRAMe<n>:WORD<o>:MISO?....1719
BUS<m>:SPI:FRAMe<n>:WORD<o>:FMISo?....1719
BUS<m>:SPI:FRAMe<n>:WORD<o>:MOSI?....1720
BUS<m>:SPI:FRAMe<n>:WORD<o>:FMOSi?....1720

BUS<m>:SPI:FRAMe<n>:DATA?

Returns the data words of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameData> Comma-separated sequence of integer values (N, L1, R1,…, LN, RN). N is the number of word pairs in the frame, and {L1,R1} …{LN,RN} are the value pairs. The values Lx and Rx are associated with the MOSI and the MISO channel, respectively. If a channel is disabled, an empty value is returned.

Example:

`BUS:SPI:FRAMe3:DATA?`
`<-- 2,10,108,35,70` (MOSI+MISO)
`2,10,,35,` (MOSI only)
`2,,108,,70` (MISO only)

Usage: Query only

BUS<m>:SPI:FCOunt?
BUS<m>:SPI:FRAMe<n>:COUNt?

Returns the number of decoded frames.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *

Return values:

<Count> Total number of decoded frames.

Usage: Query only

BUS<m>:SPI:FRAMe<n>:BITRate?

Returns the primary bit rate.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<PrimaryBitRate>
Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

BUS<m>:SPI:FRAMe<n>:STATus?

Returns the overall state of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameState> OK | VOID | INCFirst | INCLast | INSufficient

OK: the frame is valid.
VOID: the frame is empty.
INCFirst: INComplete First word. The first word does not have the expected word length.
INCLast: INComplete Last word. The last word does not have the expected word length.
INSufficient: The frame is not completely contained in the acquisition. The acquired part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:SPI:FRAMe<n>:STARt?

Returns the start time of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameStart> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:SPI:FRAMe<n>:STOP?

Returns the end time of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameStop> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:SPI:FRAMe<n>:WCOunt?

Returns the number of words in the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<WordCount> Range: 0 to 4096
*RST: 0

Usage: Query only

BUS<m>:SPI:FRAMe<n>:WORD<o>:STARt?

Returns the start time of the specified data word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the word number.

Return values:

<Start> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:SPI:FRAMe<n>:WORD<o>:STOP?

Returns the end time of the specified data word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the word number.

Return values:
<Stop> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:SPI:FRAMe<n>:WORD<o>:MISO?

Returns the data value of the specified word on the MISO line.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the word number.

Return values:
<MISOValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.
Range: 0 to 4294967295
*RST: 0

Usage: Query only

BUS<m>:SPI:FRAMe<n>:WORD<o>:FMISo?

Returns the formatted value of the specified word on the MISO line.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the word number.

Return values:
<FormattedMISOVal>

Usage: Query only

BUS<m>:SPI:FRAMe<n>:WORD<o>:MOSI?

Returns the data value of the specified word on the MOSI line.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the word number.

Return values:

<MOSIValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: 0 to 4294967295
*RST: 0

Usage: Query only

BUS<m>:SPI:FRAMe<n>:WORD<o>:FMOSi?

Returns the formatted value of the specified word on the MOSI line.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the word number.

Return values:

<FormattedMOSIVal>

Usage: Query only

24.17.4.4 SPI search settings

In search setup commands, you must specify the <SearchName> parameter. It is a string parameter that contains the search definition name. All commands are similar to SPI trigger commands.

SEARch:TRIGger:SPI:TYPE 1721
SEARch:TRIGger:SPI:FCONdition 1721
SEARch:TRIGger:SPI:MISopattern 1722
SEARch:TRIGger:SPI:MOSipattern 1722
SEARch:TRIGger:SPI:DPOPerator 1722

SEARch:TRIGger:SPI:DPOSition....1722
SEARch:TRIGger:SPI:DPTO....1723
SEARch:TRIGger:SPI:PALignment....1723

SEARch:TRIGger:SPI:TYPE <SearchName>,<Type>
SEARch:TRIGger:SPI:TYPE? <SearchName>

Sets the event to be searched for.

Parameters:

<Type> SSACtive | TIMeout | MOSI | MISO | MOMI

SSACtive
Searches for the start of the frame when slave select signal SS changes to the active state. This type is available if the slave select line is configured in the bus setup, and `BUS<m>:SPI:FRCondition` is `SS`.

TIMeout
Searches for the start of the frame when the clock idle time exceeds the timeout. This type is available if the slave select line is configured in the bus setup, and `BUS<m>:SPI:FRCondition` is `CLKTimeout`.

MOSI | MISO
Searches for a specified data pattern expected on the MOSI line or on the MISO line, respectively.

MOMI
Searches in parallel for specified data patterns expected on the MOSI and MISO lines.

*RST: SSACtive

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

SEARch:TRIGger:SPI:FCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:SPI:FCONdition? <SearchName>

Selects the operator for the data pattern: equal or not equal.

Parameters:

<DataOperator> EQUal | NEQual

*RST: EQUal

Parameters for setting and query:
<SearchName>

Firmware/software: FW 3.30

SEARch:TRIGger:SPI:MISopattern <SearchName>,<MISOPattern>
SEARch:TRIGger:SPI:MISopattern? <SearchName>
SEARch:TRIGger:SPI:MOSipattern <SearchName>,<MOSIPattern>
SEARch:TRIGger:SPI:MOSipattern? <SearchName>

Specifies a data pattern for the MISO or MOSI line, respectively.

Parameters:
<MISOPattern>
<MOSIPattern>
Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

Firmware/software: FW 3.30

SEARch:TRIGger:SPI:DPOPerator <SearchName>,<DataPosOperator>
SEARch:TRIGger:SPI:DPOPerator? <SearchName>

Operator for the data position. You can define an exact position, a position range, or let the position undefined (ANY).

Parameters:
<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe
ANY = OFF, INRange = RANGe
*RST: ANY

Parameters for setting and query:
<SearchName>

Firmware/software: FW 3.30

SEARch:TRIGger:SPI:DPOSition <SearchName>,<DataPosition>
SEARch:TRIGger:SPI:DPOSition? <SearchName>

Sets the number of bits or words before the first word of interest, see also `SEARch:TRIGger:SPI:PALignment`. These offset bits/words are skipped. The index 0 is associated with the first data bit or word.

If the position operator defines a range, also define the last bit/word of interest using `SEARch:TRIGger:SPI:DPTO`

Parameters:
<DataPosition>
Range: 0 to 32767
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

Firmware/software: FW 3.30

SEARch:TRIGger:SPI:DPTO <SearchName>,<DataPositionTo>
SEARch:TRIGger:SPI:DPTO? <SearchName>

Sets the the end value of a data postion range.

Parameters:
<DataPositionTo> Range: 1 to 32767
Increment: 1
*RST: 1

Parameters for setting and query:
<SearchName>

Firmware/software: FW 3.30

SEARch:TRIGger:SPI:PALignment <SearchName>,<DataAlignment>
SEARch:TRIGger:SPI:PALignment? <SearchName>

Defines how the specified data pattern is searched.

Parameters:
<DataAlignment> WORD | BIT

WORD
The pattern is matched only at word boundaries.

BIT
Bit-by-bit: the pattern can start at any position in the message.

*RST: WORD

Parameters for setting and query:
<SearchName>

Usage: Asynchronous command

Firmware/software: FW 3.30

24.17.4.5 SPI search results

The search on decoded SPI data returns the same results as the queries for decode results.

In search result commands, you must specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

The suffix FRAMe<m> indicates the frame index. The suffix WORD<n> indicates the word index inside a frame-

For a description of the returned values, see the corresponding commands in Chapter 24.17.4.3, "SPI decode results", on page 1715.

SEARch:RESult:SPI:FCOunt?....1724
SEARch:RESult:SPI:FRAMe<m>:COUNt?....1724
SEARch:RESult:SPI:FRAMe<m>:DATA?....1724
SEARch:RESult:SPI:FRAMe<m>:STARt?....1724
SEARch:RESult:SPI:FRAMe<m>:STATus?....1725

SEARch:RESult:SPI:FRAMe<m>:STOP?.....1725
SEARch:RESult:SPI:FRAMe<m>:WCOunt?.....1725
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:MISO?.....1726
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:MOSI?.....1726
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:STARt?.....1726
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:STOP?.....1727

SEARch:RESult:SPI:FCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:

<Count>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:COUNt?

Returns the number of frames that have matched the search criteria. In the search result table on the display, the number of rows is the number of frames that match the search criteria.

Suffix:
<m> *

Return values:
<Count>

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:DATA? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameData>

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<FrameStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:STATus? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<FrameState> OK | VOID | INCFirst | INCLast | INSufficient
*RST: OK

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:STOP? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<FrameStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:WCOunt? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<FrameWordCount> Range: 0 to 4096
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:WORD<n>:MISO? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<WordMISOValue> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:WORD<n>:MOSI? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<WordMOSIValue> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:WORD<n>:STARt? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<FrameWordStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:SPI:FRAMe<m>:WORD<n>:STOP? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<FrameWordStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

24.17.5 UART/RS-232/RS-422/RS-485 (option R&S RTO6-K510)

- Configuration....1727
- Trigger....1733
- Decode results....1735

24.17.5.1 Configuration

BUS<m>:UART:RX:SOURce....1727
BUS<m>:UART:TX:SOURce....1728
BUS<m>:UART:RX:THReshold....1728
BUS<m>:UART:TX:THReshold....1728
BUS<m>:UART:TECHnology....1729
BUS<m>:UART:BITRate....1729
BUS<m>:UART:BAUDrate....1730
BUS<m>:UART:PARity....1730
BUS<m>:UART:POLarity....1730
BUS<m>:UART:SBIT....1731
BUS<m>:UART:SSIZe....1731
BUS<m>:UART:PACKets....1731
BUS<m>:UART:TOUT....1732
BUS<m>:UART:EWORd....1732

BUS<m>:UART:RX:SOURce <RxSource>

Selects the input channel for the receiver signal.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:

<RxSource> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for RX and TX lines.
See Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: None

Usage: Asynchronous command

BUS<m>:UART:TX:SOURce <TxSource>

Selects the input channel for the transmitter signal.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<TxSource> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for RX and TX lines.
See Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C1W1

Usage: Asynchronous command

BUS<m>:UART:RX:THReshold <RxThreshold>

Sets a user-defined threshold value for the Rx line.

Alternatively, you can set the threshold according to the signal technology with `BUS<m>:UART:TECHnology`.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<RxThreshold> User-defined clock threshold

Range: -15 to 15
Increment: 0.1
*RST: 0
Default unit: V

BUS<m>:UART:TX:THReshold <TxThreshold>

Sets a user-defined threshold value for the Tx line.

Alternatively, you can set the threshold according to the signal technology with `BUS<m>:UART:TECHnology`.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<TxThreshold> User-defined clock threshold

Range: -15 to 15
Increment: 0.1
*RST: 0
Default unit: V

BUS<m>:UART:TECHnology <Technology>

Sets the threshold voltage Tx and Rx lines as defined for various signal technologies.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Technology> V15 | V25 | V165 | V125 | V09 | VM13 | V38 | V20 | V0 | MANual

V15 | V25 | V165 | V125 | V09 | V38 | V20 | V0
1.5 V, 2.5 V, 1.65 V ... respectively

VM13
-1.3 V (negative value)

MANual
Manual setting of user-defined values with `BUS<m>:UART:RX:THReshold` and `BUS<m>:UART:TX:THReshold`.

*RST: V165

BUS<m>:UART:BITRate <Bitrate>

Sets the number of transmitted bits per second.

Suffix:
<m> 1..4

Parameters:
<Bitrate>

Range: 300 to 20000000
Increment: 1
*RST: 9600
Default unit: bps

Usage: Asynchronous command

BUS<m>:UART:BAUDrate <Bitrate>

Same as `BUS<m>:UART:BITRate`.

Suffix:
<m> 1..4

Parameters:
<Bitrate> Range: 300 to 20000000
Increment: 1
*RST: 9600
Default unit: bps

Usage: Asynchronous command

BUS<m>:UART:PARity <Parity>

Defines the optional parity bit that is used for error detection.

See also: "Parity" on page 523.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Parity> NONE | ODD | EVEN | MARK | SPC | DC

MARK
The parity bit is always a logic 1.

SPC
SPaCe: The parity bit is always a logic 0.

DC
Don't Care: the parity is ignored.

*RST: NONE

Usage: Asynchronous command

BUS<m>:UART:POLarity <Polarity>

Defines the idle state of the bus. The idle state corresponds to a logic 1. The transmitted data on the bus is high (high = 1) or low (low = 1) active.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Polarity> IDLLow | IDLHigh

*RST: IDLHigh

Usage: Asynchronous command

BUS<m>:UART:SBIT <StopBits>

Sets the number of stop bits: 1; 1.5 or 2 stop bits are possible.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<StopBits> B1 | B15 | B2
*RST: B1

Usage: Asynchronous command

BUS<m>:UART:SSIZe <DataBits>

Sets the number of data bits in a message.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<DataBits> Number of data bits. 9 data bits are only possible with parity = none.
Range: 5 to 9
Increment: 1
*RST: 8

Usage: Asynchronous command

BUS<m>:UART:PACKets <Packets>

Defines the method of packet separation. A packet is a number of subsequent words in a date stream.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Packets> NONE | EWORd | TOUT

NONE
Packets are not considered.

EWORd
End word, the end condition of a packet is a pattern. To define the end word, use `BUS<m>:UART:EWORd`

TOUT
Defines a timeout between the packets. To set the timeout, use `BUS<m>:UART:TOUT`

*RST: NONE

Firmware/software: FW 2.25

BUS<m>:UART:TOUT <InterframeTime>

Sets the timeout between packets in a UART data stream. A new packet starts with the first start bit after the timeout.

The command is relevant if `BUS<m>:UART:PACKets` is set to `TOUT`.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<InterframeTime> Range: 1E-6 to 1
Increment: 1
*RST: 1E-3
Default unit: s

Usage: Asynchronous command

Firmware/software: FW 2.25

BUS<m>:UART:EWORd <EndOfFrame>

Sets the end pattern of the packets. A new packet starts with the first start bit after the defined end pattern.

The command is relevant if `BUS<m>:UART:PACKets` is set to `EWORd`.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<EndOfFrame> End word value in decimal format (range 0 to 255) or hexadecimal format (prefix #H). The query always returns hexadecimal values.

Example:
```
:BUS:UART:PACK EWOR
:BUS:UART:EWOR 10 // Decimal value
:BUS:UART:EWOR?
#H0A // Query returns hex
```

Example:
```
:BUS:UART:PACK EWOR
:BUS:UART:EWOR #Hff // Hexadecimal, prefix #H
:BUS:UART:EWOR?
#HFF
```

Usage: Asynchronous command

Firmware/software: FW 2.25

24.17.5.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:UART:TYPE....1733
TRIGger<m>:UART:SOURce....1733
TRIGger<m>:UART:DPOPerator....1733
TRIGger<m>:UART:DPOSition....1734
TRIGger<m>:UART:DPTO....1734
TRIGger<m>:UART:FCONdition....1734
TRIGger<m>:UART:DATA....1734

TRIGger<m>:UART:TYPE <Type>

Selects the trigger type for UART analysis.

See also: "Type" on page 525

Parameters:

<Type> STBT | PCKS | DATA | PRER | BRKC | STPerror

STBT: Start bit
PCKS: Packet start
DATA: Serial pattern
PRER: Parity error
BRKC: Break condition
STPerror: Stop error

*RST: STBT

Usage: Asynchronous command

TRIGger<m>:UART:SOURce <Source>

Selects the transmitter or receiver line as trigger source.

Parameters:

<Source> TX | RX

*RST: TX

Usage: Asynchronous command

TRIGger<m>:UART:DPOPerator <DataPosOperator>

Sets the operator for the data position. You can defined an exact position, or a position range.

Parameters:

<DataPosOperator> EQUal | GETHan | INRange | RANGe

INRange = RANGe

*RST: GETHan

Usage: Asynchronous command

TRIGger<m>:UART:DPOSition <DataPosition>

Sets the number of words before the first word of interest. These offset words are ignored.

Parameters:

<DataPosition> Number of words

Range: 0 to 32767
Increment: 1
*RST: 0

Usage: Asynchronous command

TRIGger<m>:UART:DPTO <DataPositionTo>

Defines the last word of interest, if `TRIGger<m>:UART:DPOPerator` defines a position range.

Parameters:

<DataPositionTo> Range: 0 to 32767
Increment: 1
*RST: 0

Usage: Asynchronous command

TRIGger<m>:UART:FCONdition <DataOperator>

Selects the operator for the data pattern (`TRIGger<m>:UART:DATA`).

Parameters:

<DataOperator> EQUal | NEQual

*RST: EQUal

TRIGger<m>:UART:DATA <Data>

Specifies the data pattern to be found on the specified trigger source, in binary or hex format. Enter the words in msb first bit order.

Parameters:

<Data> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Usage: Asynchronous command

24.17.5.3 Decode results

BUS<m>:UART:WORD<n>:RXValue?.....1735
BUS<m>:UART:WORD<n>:TXValue?.....1735
BUS<m>:UART:WORD<n>:BITRate?.....1735
BUS<m>:UART:WORD<n>:COUNt?.....1735
BUS<m>:UART:WORD<n>:SOURce?.....1736
BUS<m>:UART:WORD<n>:STARt?.....1736
BUS<m>:UART:WORD<n>:STATe?.....1736

BUS<m>:UART:WORD<n>:RXValue?
BUS<m>:UART:WORD<n>:TXValue?

Returns the value of the specified word on the Rx line or Tx line, respectively.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<Value> To set the value format, use `FORMat:BPATtern`.
The stated values for range, increment and reset are decimal values.

Range: 0 to 511
*RST: 0

Usage: Query only

BUS<m>:UART:WORD<n>:BITRate?

Returns the primary bit rate.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
The suffix is irrelevant.

Return values:

<PrimaryBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

BUS<m>:UART:WORD<n>:COUNt?

Returns the number of words in the acquisition.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
The suffix is irrelevant.

Return values:

<Count> Number of words

Usage: Query only

BUS<m>:UART:WORD<n>:SOURce?

Returns the line on which the specified word was transferred.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<WordSource> TX | RX
*RST: TX

Usage: Query only

BUS<m>:UART:WORD<n>:STARt?

Returns the start time of the specified word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<WordStart> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:UART:WORD<n>:STATe?

Returns the status of the specified word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:
<WordState> OK | FRSTart | FRENd | FRME | BREak | STERror | SPERror | PRERror | INSufficient

OK: the frame is valid.
BREak:stop bit error with 0x00 word
STERror: StarT ERror, incorrect start bit
SPERror: StoP ERror, incorrect stop bit
PRERror:PaRity ERror, incorrect parity bit.
INSufficient: The frame is not completely contained in the acquisition. The acquired part of the frame is valid.

*RST: OK

Usage: Query only

24.17.6 Audio signals (option R&S RTO6-K510)

- Configuration....1737
- Trigger....1743
- Decode results....1747
- Track and trend....1750

24.17.6.1 Configuration

BUS<m>:I2S:AVARiant....1737
BUS<m>:I2S:CLOCk:SOURce....1738
BUS<m>:I2S:CLOCk:POLarity....1738
BUS<m>:I2S:WSELect:SOURce....1738
BUS<m>:I2S:WSELect:POLarity....1739
BUS<m>:I2S:DATA:SOURce....1739
BUS<m>:I2S:DATA:POLarity....1740
BUS<m>:I2S:TCOupling....1740
BUS<m>:I2S:CLOCk:THReshold....1740
BUS<m>:I2S:WSELect:THReshold....1740
BUS<m>:I2S:DATA:THReshold....1741
BUS<m>:I2S:CHANnel:ORDer....1741
BUS<m>:I2S:WLENgth....1741
BUS<m>:I2S:BORDer....1742
BUS<m>:I2S:CHANnel:OFFSet....1742
BUS<m>:I2S:CHANnel:TDMCount....1742
BUS<m>:I2S:FOFFset....1742
BUS<m>:I2S:CHANnel:LENGth....1743

BUS<m>:I2S:AVARiant <AudioVariant>

Selects the audio signal type.

For details, see Chapter 13.5.2.1, "Audio signal configuration", on page 531.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<AudioVariant> I2S | LJ | RJ | TDM

I2S: Inter-IC Sound standard audio format.
LJ: left-justified data format
RJ: right-justified data format
TDM: Time Division Multiplexed audio format

*RST: I2S

BUS<m>:I2S:CLOCk:SOURce <ClockSource>

Selects the source of the clock line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<ClockSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital channels require installation of R&S RTO6-B1. Digital and analog channels cannot be used at the same time. For triggering on a serial bus, analog or digital input channels are required.

*RST: C1W1

BUS<m>:I2S:CLOCk:POLarity <BitClockEdge>

Sets the polarity of the clock signal, that is the edge at which the instrument samples the data on the data line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<BitClockEdge> FALLing | RISing

*RST: RISing

BUS<m>:I2S:WSELect:SOURce <WSsource>

Selects the source of the word select line for I²S standard, left- und right-justified data formats, or the source of the frame synchronization pulse for TDM audio signals.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<WSsource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital channels require installation of R&S RTO6-B1. Digital and analog channels cannot be used at the same time. For triggering on a serial bus, analog or digital input channels are required.

*RST: C2W1

BUS<m>:I2S:WSELect:POLarity <WSPolarity>

For a word select line, the polarity defines the signal values assigned to the left and right channels.

For an FSYNC line (TDM), the polarity defines the edge of the FSYNC pulse that identifies the beginning of a frame.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<WSPolarity> NORMal | INVert

NORMal: 0 = left, 1 = right channel; or rising edge for TDM
INVert: 1= left, 0 = right channel; or falling edge for TDM

*RST: NORMal

BUS<m>:I2S:DATA:SOURce <DataSource>

Selects the source of the audio data line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<DataSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital channels require installation of R&S RTO6-B1.Digital and analog channels cannot be used at the same time. For triggering on a serial bus, analog or digital input channels are required.

*RST: C3W1

BUS<m>:I2S:DATA:POLarity <SDataPolarity>

Defines the interpretation of high and low signal states on the data line.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<SDataPolarity> ACTLow | ACTHigh
ACTHigh: HIGH = 1 and LOW = 0
ACTLow: HIGH = 0 and LOW = 1
*RST: ACTHigh

BUS<m>:I2S:TCOupling <Coupling>

Sets all thresholds to the value of the clock threshold `BUS<m>:I2S:CLOCk:THReshold`.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Coupling> ON | OFF
*RST: ON

BUS<m>:I2S:CLOCk:THReshold <SCLKThreshold>

Sets the threshold value for the clock line SCLK.

If `BUS<m>:I2S:TCOupling` is `ON`, the command sets the threshold for all lines.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<SCLKThreshold> Range: -10 to 10
Increment: 1E-3
*RST: 1.6
Default unit: V

BUS<m>:I2S:WSELect:THReshold <WSThreshold>

Sets the threshold value for the word select and FSYNC lines.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:

<WSThreshold>	Range:	-10 to 10
	Increment:	1E-3
	*RST:	1.6
	Default unit:	V

BUS<m>:I2S:DATA:THReshold <SDATAThreshold>

Sets the threshold value for the data line.

Suffix:

<m>	1..4
	Selects the serial bus.

Parameters:

<SDATAThreshold>	Range:	-10 to 10
	Increment:	1E-3
	*RST:	1.6
	Default unit:	V

BUS<m>:I2S:CHANnel:ORDer <ChannelOrder>

Defines if the left or the right channel is the first channel in the frame.

The setting is not available for TDM audio signals.

Suffix:

<m>	1..4
	Selects the serial bus.

Parameters:

<ChannelOrder>	LFIRst | RFIRst	
	Left channel first or right first	
	*RST:	LFIRst

BUS<m>:I2S:WLENgth <WordLength>

Defines the number of bits in an audio data word.

Suffix:

<m>	1..4
	Selects the serial bus.

Parameters:

<WordLength>	Range:	4 to 32
	Increment:	4
	*RST:	8
	Default unit:	bit

BUS<m>:I2S:BORDer <BitOrder>

Sets the bit order of the audio data words.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<BitOrder> LSBF | MSBF
LSB first or MSB first
*RST: MSBF

BUS<m>:I2S:CHANnel:OFFSet <ChannelOffset>

Sets the number of bits between the channel start and the start of the audio word.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ChannelOffset> Range: 0 to 32 (left-justified). TDM: maximum delay is Channel length - Word length
Increment: 1
*RST: 0
Default unit: bit

BUS<m>:I2S:CHANnel:TDMCount <ChannelsTDM>

Sets the number of channels transmitted on the TDM audio line.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ChannelsTDM> Range: 1 to 8
Increment: 1
*RST: 1

BUS<m>:I2S:FOFFset <FrameOffsetTDM>

Sets the number of bits between the frame start and the start of the first channel of a TDM audio line. Each FSYNC edge restarts the offset count.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:

<FrameOffsetTDM>	Range:	0 to 256
	Increment:	1
	*RST:	0
	Default unit:	bit

BUS<m>:I2S:CHANnel:LENGth <ChlgthTDM>

Sets the number of bits in a TDM channel block.

Suffix:

<m>	1..4 Selects the serial bus.

Parameters:

<ChlgthTDM>	Range:	4 to 32
	Increment:	4
	*RST:	8
	Default unit:	bit

24.17.6.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:I2S:TYPE 1743
TRIGger<m>:I2S:TCONdition<n>:CHANnel 1744
TRIGger<m>:I2S:TCONdition<n>:CONDtion 1745
TRIGger<m>:I2S:TCONdition<n>:DMIN 1746
TRIGger<m>:I2S:TCONdition<n>:DMAX 1746
TRIGger<m>:I2S:SOWords 1746
TRIGger<m>:I2S:WSSLope 1747

TRIGger<m>:I2S:TYPE <Type>

Selects the trigger type for audio signal analysis.

Parameters:

<Type>	DATA | WINDow | CONDition | WSELect | ECONdition

DATA
Triggers on a data word or data range on a specified channel or on any channel.
To set the channel, use `TRIGger<m>:I2S:TCONdition<n>:CHANnel`.
To set the data condition, use:
`TRIGger<m>:I2S:TCONdition<n>:CONDtion`, `TRIGger<m>:I2S:TCONdition<n>:DMIN` and `TRIGger<m>:I2S:TCONdition<n>:DMAX`.

WINDow
Triggers if the decoded data values stay inside a "window" that is formed by a data range and a time specified by a number of subsequent words. It considers a selected channel or all channels.
To set up a window trigger, you define the channel and data condition in the same way as for DATA trigger type. Additionally, you set the time limit with `TRIGger<m>:I2S:SOWords`.

CONDition
The frame condition trigger sets the trigger on an AND combination of data conditions on different channels. The instrument triggers if all conditions are met inside one frame.
To set up a CONDition trigger, you define up to four channel and data conditions in the same way as for DATA trigger type.

WSELect
WordSELect: Triggers on the selected edge of the WS line (I²S standard, left- and right-justified). For TDM signals, it triggers on the selected edge of the FSYNC line. Set the edge with `TRIGger<m>:I2S:WSSLope`.

ECONdition
ErrorCONdition: Triggers on irregularities between the WS or FSYNC edges.

*RST: DATA

Usage: Asynchronous command

TRIGger<m>:I2S:TCONdition<n>:CHANnel <Channel>

Selects the audio channel on which the instrument looks for the specified data condition.

Suffix:

<n> 1..4
1 if trigger type is DATA or WINDow
Specifies the condition number if trigger type is CONDition:
– 1 | 2 for I²S standard, left- und right-justified data formats
– 1 | 2 | 3 | 4 for TDM signals

Parameters:

<Channel> ANY | TDMC1 | TDMCh1 | TDMC2 | TDMCh2 | TDMC3 | TDMCh3 | TDMC4 | TDMCh4 | TDMC5 | TDMCh5 | TDMC6 | TDMCh6 | TDMC7 | TDMCh7 | TDMC8 | TDMCh8 | LEFT | RIGHT | RIGHt

ANY
The instrument triggers on any channel on which the specified data is found.

LEFT | RIGHT = RIGHt
Available for I²S Standard, left- und right-justified data formats.

TDMCh1 | TDMCh2 | TDMCh3 | TDMCh4 | TDMCh5 | TDMCh6 | TDMCh7 | TDMCh8
Available for TDM audio signals
TDMC1 = TDMCh1, TDMC2 = TDMCh2, TDMC3 = TDMCh3, TDMC4 = TDMCh4, TDMC5 = TDMCh5, TDMC6 = TDMCh6, TDMC7 = TDMCh7, TDMC8 = TDMCh8. Query returns short form.
Note: Available audio channels depend on the configuration of the audio bus. The command `BUS<m>:I2S:CHANnel:TDMCount` specifies the number of channels in a TDM frame.

*RST: ANY

TRIGger<m>:I2S:TCONdition<n>:CONDtion <DataCondition>

Sets the operator to set a specific data pattern or a data pattern range.

Suffix:

<n> 1..4
1 if trigger type is DATA or WINDow
Specifies the condition number if trigger type is CONDition:
– 1 | 2 for I²S standard, left- und right-justified data formats
– 1 | 2 | 3 | 4 for TDM signals

Parameters:

<DataCondition> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
No range is defined.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with `TRIGger<m>:I2S:TCONdition<n>:DMIN`.

INRange | OORange
In range, Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:I2S:TCONdition<n>:DMIN` and `TRIGger<m>:I2S:TCONdition<n>:DMAX`.

*RST: OFF

TRIGger<m>:I2S:TCONdition<n>:DMIN <DataMinPattern>

Specifies a data pattern, or sets the the start value of a data pattern range.

Suffix:
<n> 1..4
1 if trigger type is DATA or WINDow
Specifies the condition number if trigger type is CONDition:
– 1 | 2 for I²S standard, left- und right-justified data formats
– 1 | 2 | 3 | 4 for TDM signals

Parameters:
<DataMinPattern> Numeric pattern in 2's complement format. See also: Chapter 24.4.6, "Bit pattern parameter", on page 1286
X bits are not allowed. If the bit string is shorter than the word length, the rightmost bit of the input bit string is aligned to the rightmost (LSB) bit of the word.

TRIGger<m>:I2S:TCONdition<n>:DMAX <DataMaxPattern>

Sets the the end value of an data range if the operator `TRIGger<m>:I2S:TCONdition<n>:CONDtion` is set to `INRange` or `OORange`.

Suffix:
<n> 1..4
1 if trigger type is DATA or WINDow
Specifies the condition number if trigger type is CONDition:
– 1 | 2 for I²S standard, left- und right-justified data formats
– 1 | 2 | 3 | 4 for TDM signals

Parameters:
<DataMaxPattern> Numeric pattern in 2's complement format. See also: Chapter 24.4.6, "Bit pattern parameter", on page 1286
DMAX must be greater or equal than DMIN.
X bits are not allowed. If the bit string is shorter than the word length, the rightmost bit of the input bit string is aligned to the rightmost (LSB) bit of the word.

TRIGger<m>:I2S:SOWords <SequenceOfWords>

Sets the number of words that is used as time limit for the "Window" trigger type. The instrument triggers if the data condition is fulfilled on the same channel for the given number of subsequent frames.

Parameters:

<SequenceOfWords> Range: 1 to 1000000
Increment: 1
*RST: 1
Default unit: word

Usage: Asynchronous command

TRIGger<m>:I2S:WSSLope <WSSlope>

Sets the edge of the WS or FSYNC signal as trigger condition. The instrument triggers on the first clock edge after the specified edge.

Parameters:

<WSSlope> POSitive | NEGative
*RST: POSitive

Usage: Asynchronous command

24.17.6.3 Decode results

BUS<m>:I2S:FCOunt? 1747
BUS<m>:I2S:FRAMe<n>:STATe? 1747
BUS<m>:I2S:FRAMe<n>:STARt? 1748
BUS<m>:I2S:FRAMe<n>:STOP? 1748
BUS<m>:I2S:FRAMe<n>:LEFT:VALue? 1748
BUS<m>:I2S:FRAMe<n>:RIGHt:VALue? 1748
BUS<m>:I2S:FRAMe<n>:LEFT:STATe? 1749
BUS<m>:I2S:FRAMe<n>:RIGHt:STATe? 1749
BUS<m>:I2S:FRAMe<n>:TDM<o>:STATe? 1749
BUS<m>:I2S:FRAMe<n>:TDM<o>:VALue? 1749

BUS<m>:I2S:FCOunt?

Returns the number of decoded frames.

Suffix:

<m> 1..4
Selects the serial bus.

Return values:

<Count> Number of decoded audio frames

Usage: Query only

BUS<m>:I2S:FRAMe<n>:STATe?

Returns the overall state of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<FrameState> ERRor | OK | INSufficient
OK: the frame is valid.
ERRor: an error occured in the frame.
INSufficient: The frame is not completely contained in the acquisition. The acquired part of the frame is valid.

Usage: Query only

BUS<m>:I2S:FRAMe<n>:STARt?
BUS<m>:I2S:FRAMe<n>:STOP?

Return the start time and stop time of the selected frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<Start>, <Stop> Range: -100E+24 to 100E+24
*RST: 0

Usage: Query only

BUS<m>:I2S:FRAMe<n>:LEFT:VALue?
BUS<m>:I2S:FRAMe<n>:RIGHt:VALue?

Return the data values of the left and right channel, respectively.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<Value> Comma-separated list of values. To set the value format, use `FORMat:BPATtern`.
Range: 0 to 4294967295
*RST: 0

Usage: Query only

BUS<m>:I2S:FRAMe<n>:LEFT:STATe?
BUS<m>:I2S:FRAMe<n>:RIGHt:STATe?

Return the status of the left and right channel of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<WordState> ERRor | OK | INSufficient

OK: the channel data is valid.
ERRor: an error occured in the channel.
INSufficient: the channel is not completely contained in the acquisition.

Usage: Query only

BUS<m>:I2S:FRAMe<n>:TDM<o>:STATe?

Returns the state of the indicated channel of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> 1..8
Selects the TDM channel.

Return values:

<State> ERRor | OK | INSufficient

OK: the channel data is valid.
ERRor: an error occured in the channel.
INSufficient: the channel is not completely contained in the acquisition.

*RST: UNDefined

Usage: Query only

BUS<m>:I2S:FRAMe<n>:TDM<o>:VALue?

Returns the data value of the indicated TDM channel.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> 1..8
Selects the TDM channel.

Return values:

<Value> To set the value format, use `FORMat:BPATtern` on page 1294.
The stated values for range, increment and reset are decimal values.

Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

24.17.6.4 Track and trend

BUS<m>:I2S:TRACk:LEFT 1750
BUS<m>:I2S:TRACk:RIGHt 1750
BUS<m>:I2S:TRACk:TD1Ch 1750
BUS<m>:I2S:TRACk:TD2Ch 1750
BUS<m>:I2S:TRACk:TD3Ch 1750
BUS<m>:I2S:TRACk:TD4Ch 1750
BUS<m>:I2S:TRACk:TD5Ch 1751
BUS<m>:I2S:TRACk:TD6Ch 1751
BUS<m>:I2S:TRACk:TD7Ch 1751
BUS<m>:I2S:TRACk:TD8Ch 1751
MEASurement<m>:TRACk[:STATe] 1751
MEASurement<m>:TRACk:DATA:HEADer? 1751
MEASurement<m>:TRACk:DATA:STYPe? 1751
MEASurement<m>:TRACk:DATA[:VALues]? 1752

BUS<m>:I2S:TRACk:LEFT <Left>
BUS<m>:I2S:TRACk:RIGHt <Right>

Enables or disables the track of the indicated channel. The commands are relevant for I²S standard, left-justified and right-justified audio data formats.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Left>, <Right> ON | OFF

*RST: OFF

BUS<m>:I2S:TRACk:TD1Ch <TDMCh1>
BUS<m>:I2S:TRACk:TD2Ch <TDMCh2>
BUS<m>:I2S:TRACk:TD3Ch <TDMCh3>
BUS<m>:I2S:TRACk:TD4Ch <TDMCh4>

BUS<m>:I2S:TRACk:TD5Ch <TDMCh5>
BUS<m>:I2S:TRACk:TD6Ch <TDMCh6>
BUS<m>:I2S:TRACk:TD7Ch <TDMCh7>
BUS<m>:I2S:TRACk:TD8Ch <TDMCh8>

Enables or disables the track of the indicated channel. The commands are relevant for TDM audio data.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<TDMChX> ON | OFF
*RST: OFF

MEASurement<m>:TRACk[:STATe] <State>

Enables the track functionality and displays the track.

The track functionality requires at least one option, see "Enable (Track)" on page 361.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

MEASurement<m>:TRACk:DATA:HEADer?

Returns the header of the track.

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Usage: Query only

MEASurement<m>:TRACk:DATA:STYPe?

Returns the data type: TRK (track).

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Usage: Query only

MEASurement<m>:TRACk:DATA[:VALues]?

Returns the data of track points for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

To set the export format, use `FORMat[:DATA]`.

Suffix:

<m>	1..10 See "Selection of the measurement group: MEASurement<m>" on page 1479.
Usage:	Query only

24.17.7 Custom: manchester / NRZ (option R&S RTO6-K510)

- Configuration....1752
- Trigger....1768
- Filter....1774
- Decode results....1778
- Search settings....1784
- Search results....1789

24.17.7.1 Configuration

In all `BUS<m>:CMSB` commands, the suffix <m> selects the serial bus.

In all `BUS<m>:CMSB:FRAMe<n>` commands, the suffix <n> selects a frame number, and the suffix <o> selects a cell number.

BUS<m>:CMSB:CODing.... 1753
BUS<m>:CMSB:MANChester:DATA.... 1754
BUS<m>:CMSB:MANChester:POLarity.... 1754
BUS<m>:CMSB:MANChester:THReshold:HIGH.... 1754
BUS<m>:CMSB:MANChester:THReshold:LOW.... 1754
BUS<m>:CMSB:MANChester:THReshold:PRESet.... 1755
BUS<m>:CMSB:MANChester:THReshold:COUPling.... 1755
BUS<m>:CMSB:MANChester:CPHase.... 1756
BUS<m>:CMSB:NRZ:CLCK.... 1756
BUS<m>:CMSB:NRZ:DATA.... 1756
BUS<m>:CMSB:NRZ:IDLPolarity.... 1757
BUS<m>:CMSB:NRZ:CPOLarity.... 1757
BUS<m>:CMSB:NRZ:CPHase.... 1758
BUS<m>:CMSB:NRZ:ENBLe.... 1758
BUS<m>:CMSB:NRZ:ENAPolarity.... 1759
BUS<m>:CMSB:NRZ:POLarity.... 1759
BUS<m>:CMSB:NRZ:THReshold:CLCK.... 1759
BUS<m>:CMSB:NRZ:HYSTeresis:CLCK.... 1759
BUS<m>:CMSB:NRZ:THReshold:DATA.... 1760
BUS<m>:CMSB:NRZ:HYSTeresis:DATA.... 1760
BUS<m>:CMSB:NRZ:THReshold:ENBLe.... 1760

BUS<m>:CMSB:NRZ:HYSTeresis:ENBLe 1760
BUS<m>:CMSB:NRZ:THReshold:PRESet 1761
BUS<m>:CMSB:NRZ:THReshold:COUPling 1761
BUS<m>:CMSB:BITRate:ENABle 1762
BUS<m>:CMSB:BITRate:VALue 1762
BUS<m>:CMSB:GAPTime:ENABle 1762
BUS<m>:CMSB:GAPTime:VALue 1762
BUS<m>:CMSB:ADDFrame 1763
BUS<m>:CMSB:FCOunt? 1763
BUS<m>:CMSB:CLR 1763
BUS<m>:CMSB:FRAMe<n>:TYPE 1763
BUS<m>:CMSB:FRAMe<n>:APPend 1764
BUS<m>:CMSB:FRAMe<n>:CCOunt? 1764
BUS<m>:CMSB:FRAMe<n>:CELL<o>:NAME 1764
BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITCount 1764
BUS<m>:CMSB:FRAMe<n>:CELL<o>:CONDition 1765
BUS<m>:CMSB:FRAMe<n>:CELL<o>:FORMat 1765
BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITorder 1766
BUS<m>:CMSB:FRAMe<n>:CELL<o>:CRGB 1766
BUS<m>:CMSB:EXRBits 1766
BUS<m>:CMSB:FRAMe<n>:CELL<o>:CLMN 1766
BUS<m>:CMSB:LOAD 1767
BUS<m>:CMSB:SAVE 1767

BUS<m>:CMSB:CODing <CodingStandard>

Selects the custom serial bus coding standard.

Suffix:

<m> 1..4

Parameters:

<CodingStandard> MANC | MANT | NRZ | NRZU

MANC
Manchester (normal polarity)

MANT
Manchester II (inverted polarity).
Note that some additional subtle differences between MANC and MANT require separate protocols.

NRZ
NRZ (non-return-to-zero), clocked

NRZU
NRZ (non-return-to-zero), unclocked

*RST: MANC

BUS<m>:CMSB:MANChester:DATA <SourceMANData>

Selects the source channel for the data signal. For triggering on a serial bus, analog channels "C1"–"C4" are required. Otherwise, if no serial bus trigger has been selected, permitted source selections include the mathematical channels and the reference channels .

Suffix:
<m> 1..4

Parameters:
<SourceMANData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: C1W1

BUS<m>:CMSB:MANChester:POLarity <PolarityData>

Selects the polarity of the custom serial bus data signal in Manchester coding standards.

Suffix:
<m> 1..4

Parameters:
<PolarityData> NORMal | INVert

NORMal
Manchester or Manchester II polarity remains unchanged

INVert
Manchester polarity is inverted and becomes Manchester II polarity, Manchester II polarity is inverted and becomes Manchester polarity

*RST: NORMal

BUS<m>:CMSB:MANChester:THReshold:HIGH <ThresholdUpper>

Sets the upper threshold for data signal digitization in Manchester coding standards.

Suffix:
<m> 1..4

Parameters:
<ThresholdUpper> Range: -25 to 25
Increment: 0.1
*RST: 5
Default unit: V

BUS<m>:CMSB:MANChester:THReshold:LOW <ThresholdLower>

Sets the lower threshold for data signal digitization in Manchester coding standards.

Suffix:
<m> 1..4

Parameters:
<ThresholdLower> Range: -25 to 25
Increment: 0.1
*RST: -5
Default unit: V

BUS<m>:CMSB:MANChester:THReshold:PRESet <PresetValue>

Sets the Manchester thresholds to predefined or individually definable voltage levels.

Suffix:
<m> 1..4

Parameters:
<PresetValue> V05 | V2 | V5 | V7 | MANual

V05
Sets the upper threshold to +0.5 V and the lower threshold to -0.5 V

V2
Sets the upper threshold to +2.0 V and the lower threshold to -2.0 V

V5
Sets the upper threshold to +5.0 V and the lower threshold to -5.0 V

V7
Sets the upper threshold to +7.0 V and the lower threshold to -7.0 V

MANual
Allows to set individual threshold voltage levels

*RST: V5

BUS<m>:CMSB:MANChester:THReshold:COUPling <ThresholdCoupling>

Couples the upper and lower threshold values for the Manchester and Manchester II coding standards. The values are coupled to voltages with the same magnitude but opposite sign (positive for the upper threshold and negative for the lower threshold). However, if the upper threshold is set to a negative voltage or the lower threshold is set to a positive voltage, coupling is disabled, and the other voltage (the one that was not actively set) is automatically adjusted, to avoid an upper threshold below the lower one, or a lower threshold above the upper one.

Suffix:
<m> 1..4

Parameters:
<ThresholdCoupling> ON | OFF

ON
Activates coupling of the upper and lower threshold values.

OFF
Disables coupling of the upper and lower threshold values.

*RST: ON

BUS<m>:CMSB:MANChester:CPHase <ClockPhase>

Selects the phase of the custom serial bus clock signal for the "Manchester" coding standards. For details, see "Clock Phase (Manchester)" on page 551.

Suffix:
<m> 1..4

Parameters:
<ClockPhase> FEDGe | SEDGe

FEDGe
Sets the sampling edge to be on the first edge.

SEDGe
Sets the sampling edge to be on the second edge.

AUTO
Lets the decoder automatically select the method ("First Edge" or "Second Edge") for detecting the sampling edge.

*RST: SEDGe

BUS<m>:CMSB:NRZ:CLCK <SourceNRZClock>

Selects the source channel for the clock signal in the NRZ Clocked coding standard.

For triggering on the serial bus when the NRZ clocked coding standard is selected, analog or digital channel sources are required.

Otherwise, if no serial bus trigger has been selected, permitted source selections include the mathematical channels and the reference channels.

Suffix:
<m> 1..4

Parameters:
<SourceNRZClock> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

*RST: C2W1

BUS<m>:CMSB:NRZ:DATA <SourceNRZData>

Selects the source channel for the data signal in NRZ coding standards.

For triggering on the serial bus when the NRZ clocked coding standard is selected, analog or digital channel sources are required.

For triggering on the serial bus when the NRZ unclocked coding standard is selected, analog channel sources are required.

Otherwise, if no serial bus trigger has been selected, permitted source selections include the mathematical channels and the reference channels.

Suffix:
<m> 1..4

Parameters:
<SourceNRZData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

*RST: C1W1

BUS<m>:CMSB:NRZ:IDLPolarity <PolarityIdle>

Selects the idle polarity of the custom serial bus data signal (only available for the coding standard "NRZ Unclocked").

Suffix:
<m> 1..4

Parameters:
<PolarityIdle> IDLLow | IDLHigh

IDLLow
Sets the base value of the data bus to be "0". After an idle period, the data signal starts with a low-to-high transition

IDLHigh
Sets the base value of the data bus to be "1". After an idle period, the data signal starts with a high-to-low transition

*RST: IDLLow

BUS<m>:CMSB:NRZ:CPOLarity <PolarityClock>

Selects the polarity of the clock signal for the coding standard NRZ Clocked.

Suffix:
<m> 1..4

Parameters:
<PolarityClock> IDLLow | IDLHigh

IDLLow
Sets the base value of the clock to be "0", the clock signal starts with a low-to-high transition

IDLHigh
Sets the base value of the clock to be "1", the clock signal starts with a high-to-low transition.

*RST: IDLLow

BUS<m>:CMSB:NRZ:CPHase <ClockPhase>

Selects the phase of the custom serial bus clock signal for the coding standard "NRZ Clocked", depending on `BUS<m>:CMSB:NRZ:IDLPolarity` on page 1757.

Suffix:
<m> 1..4

Parameters:
<ClockPhase> FEDGe | SEDGe

FEDGe
Sets the clocking transaction to be on the first edge:
If Clock Polarity = "IDLLow", data are captured on the clock's rising edge (low-to-high transition) and propagated on a falling edge
If Clock Polarity = "IDLHigh", data are captured on the clock's falling edge (high-to-low transition) and propagated on a rising edge

SEDGe
Sets the clocking transaction to be on the second edge:
If Clock Polarity = "IDLLow", data are captured on the clock's falling edge (high-to-low transition) and propagated on a rising edge
If Clock Polarity = "IDLHigh", data are captured on the clock's rising edge (low-to-high transition) and propagated on a falling edge

*RST: FEDGe

BUS<m>:CMSB:NRZ:ENBLe <SourceNRZEnable>

Selects the input source for the custom serial bus enable signal.

If an input is chosen, signals will be only decoded while this channel is in the enabled state. This allows you to mark a time when the signal on the selected source is active and when not.

For triggering on a serial bus, analog channels "C1"–"C4" are required. Otherwise, if no serial bus trigger has been selected, permitted source selections include the mathematical channels and the reference channels.

Suffix:
<m> 1..4

Parameters:
<SourceNRZEnable> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

BUS<m>:CMSB:NRZ:ENAPolarity <PolarityEnable>

Sets whether the transmitted enable signal is active when the voltage is below the threshold (`ENALow`) or higher than it (`ENAHIgh`).

Suffix:
<m> 1..4

Parameters:
<PolarityEnable> ENALow | ENAHigh

*RST: ENAHigh

BUS<m>:CMSB:NRZ:POLarity <PolarityData>

Selects the polarity of the custom serial bus data signal in NRZ coding standards.

Suffix:
<m> 1..4

Parameters:
<PolarityData> ACTLow | ACTHigh

ACTLow
Active low: the value "1" is represented by a voltage below the threshold

ACTHigh
Active high: the value "1" is represented by a voltage above the threshold

*RST: ACTHigh

BUS<m>:CMSB:NRZ:THReshold:CLCK <ThresholdClock>

Sets the threshold for the clock signal digitization in the NRZ Clocked coding standard.

Suffix:
<m> 1..4

Parameters:
<ThresholdClock> Range: -25 to 25
Increment: 0.1
*RST: 2
Default unit: V

BUS<m>:CMSB:NRZ:HYSTeresis:CLCK <HystClock>

Sets the hysteresis size on the clock channel.

Suffix:
<m> 1..4

Parameters:

<HystClock>	Range:	-25 to 25
	Increment:	0.1
	*RST:	0.1
	Default unit:	V

BUS<m>:CMSB:NRZ:THReshold:DATA <ThresholdData>

Sets the threshold for the data signal digitization in NRZ coding standards.

Suffix:

<m> 1..4

Parameters:

<ThresholdData>	Range:	-25 to 25
	Increment:	0.1
	*RST:	2
	Default unit:	V

BUS<m>:CMSB:NRZ:HYSTeresis:DATA <HystData>

Sets the hysteresis size on the data line channel.

Suffix:

<m> 1..4

Parameters:

<HystData>	Range:	-25 to 25
	Increment:	0.1
	*RST:	0.1
	Default unit:	V

BUS<m>:CMSB:NRZ:THReshold:ENBLe <ThresholdEnable>

Sets the threshold for the enable signal digitization in NRZ coding standards.

Suffix:

<m> 1..4

Parameters:

<ThresholdEnable>	Range:	-25 to 25
	Increment:	0.1
	*RST:	2
	Default unit:	V

BUS<m>:CMSB:NRZ:HYSTeresis:ENBLe <HystEnable>

Sets the hysteresis size on the enable line channel.

Suffix:

<m> 1..4

Parameters:

<HystEnable> Range: -25 to 25
Increment: 0.1
*RST: 0.1
Default unit: V

BUS<m>:CMSB:NRZ:THReshold:PRESet <PresetValue>

Sets the NRZ thresholds to predefined or individually definable voltage levels.

Suffix:

<m> 1..4

Parameters:

<PresetValue> V05 | V2 | V5 | V7 | MANual

V05
Sets the clock and data threshold to +0.5 V (in case of NRZ Unclocked: data threshold, only)

V2
Sets the clock and data threshold to +2.0 V (in case of NRZ Unclocked: data threshold, only)

V5
Sets the clock and data threshold to +5.0 V (in case of NRZ Unclocked: data threshold, only)

V7
Sets the clock and data threshold to +7.0 V (in case of NRZ Unclocked: data threshold, only)

MANual
Allows to set individual threshold voltage levels

*RST: V5

BUS<m>:CMSB:NRZ:THReshold:COUPling <ThresholdCoupling>

Couples the clock and data threshold values for the NRZ Clocked coding standard. The values are coupled to the same number.

Suffix:

<m> 1..4

Parameters:

<ThresholdCoupling> ON | OFF

ON
Activates coupling of the NRZ clock and data threshold values.

OFF
Disables coupling of the NRZ clock and data threshold values.

*RST: ON

BUS<m>:CMSB:BITRate:ENABle <BitrateEnable>

Enables the bit rate settings (must always be enabled for the coding standard "NRZ Unclocked", and also for triggering on signals in any coding standard).

Suffix:
<m> 1..4

Parameters:
<BitrateEnable> ON | OFF

ON
Bit rate settings enabled

OFF
Bit rate settings disabled

*RST: OFF

BUS<m>:CMSB:BITRate:VALue <Bitrate>

Sets the transmission speed setting for the custom serial bus data signal.

Suffix:
<m> 1..4

Parameters:
<Bitrate>
Range: 300 to 2E9
Increment: 100E3
*RST: 10E6
Default unit: bps

BUS<m>:CMSB:GAPTime:ENABle <GapTimeEnable>

Enables the gap time settings (must always be enabled for the coding standard "NRZ Unclocked", and also for triggering on signals in any coding standard).

Suffix:
<m> 1..4

Parameters:
<GapTimeEnable> ON | OFF

ON
Gap time settings enabled

OFF
Gap time settings disabled

*RST: OFF

BUS<m>:CMSB:GAPTime:VALue <MinGapTime>

Sets a minimum gap time for synchronization.

Suffix:
<m> 1..4

Parameters:

<MinGapTime>	Range:	1E-9 to 1
	Increment:	1E-9
	*RST:	10E-6
	Default unit:	s

BUS<m>:CMSB:ADDFrame

Creates an empty frame format description and adds it to the end of the frame description list.

Suffix:
<m> 1..4

Usage: Event

BUS<m>:CMSB:FCOunt?

Returns the number of frames.

Suffix:
<m> 1..4

Return values:
<Count>

Usage: Query only

BUS<m>:CMSB:CLR

Erases all cells and frames that have been created for a specific custom protocol.

Suffix:
<m> 1..4

Usage: Event

BUS<m>:CMSB:FRAMe<n>:TYPE <FrameType>

Enables the user to set a string to describe the frame type, typically according to the applicable protocol standard specifications. (For example, MDIO (option R&S RTO6-K540) defines the frames READ, WRITE, ADDRESS, etc.)

Suffix:
<m> 1..4

<n> *

Parameters:
<FrameType>

BUS<m>:CMSB:FRAMe<n>:APPend

Creates an empty cell description and adds it to the end of the active frame description.

Suffix:
<m> 1..4
<n> *

Usage: Event

BUS<m>:CMSB:FRAMe<n>:CCOunt?

Returns the number of cells in the specified frame.

Suffix:
<m> 1..4
<n> *

Return values:
<Count>

Usage: Query only

BUS<m>:CMSB:FRAMe<n>:CELL<o>:NAME <CellName>

Enables the user to set a cell name within a frame. Names do not have to be unique, they are just for user support.

Suffix:
<m> 1..4
<n> *
<o> *

Parameters:
<CellName>

BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITCount <BitCount>

Sets the bit count of a cell, hence its length. Based upon the lengths of the previous cells, this also defines the position of the cell start and end within a frame.

Suffix:
<m> 1..4
<n> *
<o> *

Parameters:

<BitCount>	Range:	1 to 64
	Increment:	1
	*RST:	1

BUS<m>:CMSB:FRAMe<n>:CELL<o>:CONDition <Condition>

Sets various operators for a cell, to identify, e.g., mandatory values such as a CRC checksum or an ID, that help to identify a frame.

The implemented conditions and functionalities are the "equal" and "array" operators.

The numeric format of the condition needs to be set according to `BUS<m>:CMSB:FRAMe<n>:CELL<o>:FORMat` on page 1765.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<Condition>

BUS<m>:CMSB:FRAMe<n>:CELL<o>:FORMat <CellFormat>

Selects the numeric data format for the command `BUS<m>:CMSB:FRAMe<n>:CELL<o>:CONDition` on page 1765, as well as for the result and honeycomb display.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<CellFormat> DEC | HEX | OCT | BIN

DEC
Decimal

HEX
Hexadecimal

OCT
Octal

BIN
Binary

*RST: BIN

BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITorder <BitOrder>

Selects in which order the bits of a cell are evaluated, as well as presented in the results table and honeycomb display.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<BitOrder> LSBF | MSBF

LSBF
Least significant bit first, evaluation starts at the LSB

MSBF
Most significant bit first, evaluation starts at the MSB

*RST: MSBF

BUS<m>:CMSB:FRAMe<n>:CELL<o>:CRGB <Color>

Selects a cell's color representation in the honeycomb display.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<Color> ARGB value of the color to be used for the table entry. ARGB=<Opacity(alpha) value><red value><green value><blue value>, in hexadecimal or decimal format.

Range: 0 to 4294967295
Increment: 1
*RST: 0

BUS<m>:CMSB:EXRBits

Exports the decoded frames as raw bits.

Suffix:

<m> 1..4

Usage: Event

BUS<m>:CMSB:FRAMe<n>:CELL<o>:CLMN <Column>

Selects which cell shall be displayed in which result column of the decode table.

The decode table supports three result columns, which have to be unique for each frame type. For different frame types, though, different result columns can be defined to display unrelated information.

Suffix:

<m> 1..4

<n> *

<o> *

Parameters:

<Column> NONE | COL1 | COL2 | COL3

NONE
The result is not displayed in the decode table.

COL1
The result is displayed in column 1 of the decode table.

COL2
The result is displayed in column 2 of the decode table.

COL3
The result is displayed in column 3 of the decode table.

*RST: NONE

BUS<m>:CMSB:LOAD <FileName>

Opens an existing frame description file in xml format. The default path is `\\Public\Documents\Rohde-Schwarz\RTx\SaveXML`

Suffix:

<m> 1..4

Setting parameters:

<FileName>

Usage: Setting only

BUS<m>:CMSB:SAVE <FileName>

Saves a created frame description into an xml file ("Save As..."). The default path is `\\Public\Documents\Rohde-Schwarz\RTx\SaveXML`

Suffix:

<m> 1..4

Setting parameters:

<FileName>

Usage: Setting only

24.17.7.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:CMSB:TYPE....1768
TRIGger<m>:CMSB:PATTern....1769
TRIGger<m>:CMSB:ICONdition....1769
TRIGger<m>:CMSB:IMIN....1769
TRIGger<m>:CMSB:IMAX....1770
TRIGger<m>:CMSB:NRZ:WRDLength....1770
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:ENABle....1770
TRIGger<m>:CMSB:ADVanced:FRENable....1770
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:ENABle....1770
TRIGger<m>:CMSB:ADVanced:FIENable....1770
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:BIT....1771
TRIGger<m>:CMSB:ADVanced:BIT....1771
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMAX....1771
TRIGger<m>:CMSB:ADVanced:DMAX....1771
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMIN....1772
TRIGger<m>:CMSB:ADVanced:DMIN....1772
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DOPerator....1772
TRIGger<m>:CMSB:ADVanced:DOPerator....1772
TRIGger<m>:CMSB:ADVanced:ERRor<n>:ENABle....1772
TRIGger<m>:CMSB:ADVanced:ERENable....1772
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMAX....1773
TRIGger<m>:CMSB:ADVanced:IMAX....1773
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMIN....1773
TRIGger<m>:CMSB:ADVanced:IMIN....1773
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IOPerator....1773
TRIGger<m>:CMSB:ADVanced:IOPerator....1773

TRIGger<m>:CMSB:TYPE <Type>

Selects the trigger type for custom serial bus analysis.

Suffix:

<m>	1..3

Parameters:

<Type>	STARt | PATTern | ADVanced

STARt
Triggers on the frame start, which is the end of the gap time as specified in `BUS<m>:CMSB:GAPTime:VALue` on page 1762

ADVanced
Advanced trigger with `TRIGger:CMSB:ADVanced` commands. Only available if frames are defined.

PATTern
Triggers on a data pattern to be specified in `TRIGger<m>:CMSB:PATTern` on page 1769

*RST: STARt

TRIGger<m>:CMSB:PATTern <DataPattern>

Sets the pattern match conditions for a payload data check. The trigger is set to the first occurrence of a matching data bit pattern (which can be freely specified), starting after the minimum gap time (`BUS<m>:CMSB:GAPTime:VALue` on page 1762), and after the detected start of the data frame.

Suffix:
<m> 1..3

Parameters:
<DataPattern>

TRIGger<m>:CMSB:ICONdition <DataIdxOperator>

Sets the operator to set a specific bit index (data position).

Suffix:
<m> 1..3

Parameters:
<DataIdxOperator> EQUal | GETHan | INRange

EQUal
Equal

GETHan
Greater than or equal

INRange
In range

*RST: INRange

TRIGger<m>:CMSB:IMIN <DataPosition>

Sets the bit index (data position). If `TRIGger<m>:CMSB:ICONdition` on page 1769 is set to "INRange", "IMIN" sets the start value of the bit index range.

Suffix:
<m> 1..3

Parameters:
<DataPosition> Range: 0 to 65535
Increment: 1
*RST: 0

TRIGger<m>:CMSB:IMAX <DataPositionTo>

Sets the end value of the bit index range (data position range), if `TRIGger<m>:CMSB:ICONdition` on page 1769 is set to "INRange".

Suffix:
<m> 1..3

Parameters:
<DataPositionTo> Range: 0 to 65535
Increment: 1
*RST: 65535

TRIGger<m>:CMSB:NRZ:WRDLength <NRZWordLength>

Sets the number of bits in an NRZ Unclocked word.

Suffix:
<m> 1..3

Parameters:
<NRZWordLength> Range: 0 to 31
Increment: 1
*RST: 8

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:ENABle <Enable>
TRIGger<m>:CMSB:ADVanced:FRENable <Frame>, <Enabler>

Enables or disables the specific frame to be triggered on.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Enabler> ON | OFF

Example: `:TRIGger:CMSB:ADVanced:FRENable "Master" 1`
Enables the Master frame.
`:TRIGger:CMSB:ADVanced:FRAMe1:ENABle 1`
Enables the 1st frame that you have defined.

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:ENABle <CondEnabler>
TRIGger<m>:CMSB:ADVanced:FIENable <Frame>, <Field>, <Enabler>

Enables or disables the specific field within the defined frame to be triggered on.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Enabler> ON | OFF

Example: `:TRIGger:CMSB:ADVanced:FIEN "Master", "CRC", 1`
Enables the CRC field of the Master frame.
`:TRIGger:CMSB:ADVanced:FRAMe1:FLD3:ENABle 1`
Enables the 3rd field of the 1st frame that you have defined.

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:BIT <BitState>
TRIGger<m>:CMSB:ADVanced:BIT <Frame>, <Field>, <Bit>

Sets the bit state of a field to be triggered on that only consists of one bit.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Bit> ONE | ZERO | DC

ONE
1

ZERO
0

DC
"Do not care" (DC) = X

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMAX <DataMax>
TRIGger<m>:CMSB:ADVanced:DMAX <Frame>, <Field>, <Data>

Sets the end value of a data pattern range if `TRIGger<m>:CMSB:ADVanced:DOPerator` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Data>

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMIN <DataMin>
TRIGger<m>:CMSB:ADVanced:DMIN <Frame>, <Field>, <Data>

Specifies the data pattern, or sets the start value of a data pattern range.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Data>

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DOPerator <DataOperator>
TRIGger<m>:CMSB:ADVanced:DOPerator <Frame>, <Field>, <Operator>

Sets the operator for the data pattern.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Operator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

ANY = OFF
The data of the required pattern is not relevant.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
This condition requires one endpoint value to be set using `TRIGger<m>:CMSB:ADVanced:DMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `TRIGger<m>:CMSB:ADVanced:DMAX` and `TRIGger<m>:CMSB:ADVanced:DMIN`.

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:ERRor<n>:ENABle <Enable>
TRIGger<m>:CMSB:ADVanced:ERENable <ErrorName>, <Enabler>

Enables triggering on errors. You can search for all error types in parallel.

Suffix:
<m> 1..3

Setting parameters:
<ErrorName>

<Enabler> ON | OFF

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMAX <IndexMax>
TRIGger<m>:CMSB:ADVanced:IMAX <Frame>, <Field>, <Data>

Sets the end value of an index range if `TRIGger<m>:CMSB:ADVanced:IMAX` is set to `INRange` or `RANGe`.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Data>

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMIN <IndexMin>
TRIGger<m>:CMSB:ADVanced:IMIN <Frame>, <Field>, <Data>

Specifies the index, or sets the start value of an index range.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Data>

Usage: Setting only

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IOPerator <IndexOperator>
TRIGger<m>:CMSB:ADVanced:IOPerator <Frame>, <Field>, <Operator>

Sets the operator for the index in the selected field of the selected frame.

Suffix:
<m> Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Setting parameters:
<Frame>

<Field>

<Operator> ANY | OFF | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

ANY = OFF
The index of the required pattern is not relevant.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
This condition requires one endpoint value to be set using `TRIGger<m>:CMSB:ADVanced:IMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `TRIGger<m>:CMSB:ADVanced:IMAX` and `TRIGger<m>:CMSB:ADVanced:IMIN`.

Usage: Setting only

24.17.7.3 Filter

BUS<m>:CMSB:FILTer:ENABle <Enable>

Enables filtering for the custom serial bus.

Suffix:
<m> 1..4

Parameters:
<Enable> ON | OFF
*RST: OFF

BUS<m>:CMSB:FILTer:ERENable <ErrorName>, <Enabler>

Enables filtering on error.

Suffix:
<m> 1..4

Setting parameters:
<ErrorName>

<Enabler> ON | OFF

Usage: Setting only

BUS<m>:CMSB:FILTer:ERRor<n>:ENABle <Enable>

Enables filtering on error. You can filter all error types in parallel.

Suffix:
<m> 1..4

<n> *

Parameters:
<Enable> ON | OFF
*RST: ON

BUS<m>:CMSB:FILTer:FRAMe<n>:ENABle <Enable>

Enables or disables the specific frame to be filtered.

Suffix:
<m> 1..4

<n> *

Parameters:
<Enable> ON | OFF
*RST: OFF

BUS<m>:CMSB:FILTer:FRENable <Frame>, <Enabler>

Enables or disables the specific frame to be filtered.

Suffix:
<m> 1..4

Setting parameters:
<Frame>

<Enabler> ON | OFF

Usage: Setting only

BUS<m>:CMSB:FILTer:FIENable <Frame>, <Field>, <Enabler>

Enables or disables the specific field within the defined frame to be filtered.

Suffix:
<m> 1..4

Setting parameters:
<Frame>

<Field>

<Enabler> ON | OFF

Usage: Setting only

BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:BIT <BitState>

Sets the bit state of a field to be filtered on that only consists of one bit.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<BitState>	ONE | ZERO | DC *RST: DC

BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:DMAX <DataMax>

Sets the end value of a data pattern range if `BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:DOPerator` is set to `INRange` or `OORange`.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<DataMax>

BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:DMIN <DataMin>

Specifies the data pattern, or sets the start value of a data pattern range.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<DataMin>

BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:DOPerator <DataOperator>

Sets the operator for the data pattern.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<DataOperator>	EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange *RST: EQUal

BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:ENABle <CondEnabler>

Enables or disables the specific field within the defined frame to be filtered.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<CondEnabler>	ON | OFF	
	*RST:	OFF

BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:IMAX <IndexMax>

Sets the end value of an index range if `BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:IOPerator` is set to `INRange` or `RANGe`.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<IndexMax>	Range:	1 to 65534
	Increment:	1
	*RST:	65534

BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:IMIN <IndexMin>

Specifies the index, or sets the start value of an index range.

Suffix:

<m>	1..4
<n>	*
<o>	*

Parameters:

<IndexMin>	Range:	1 to 65534
	Increment:	1
	*RST:	1

BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:IOPerator <IndexOperator>

Sets the operator for the index in the selected field of the selected frame.

Suffix:

<m>	1..4

<n> *

<o> *

Parameters:

<IndexOperator> EQUal | INRange | RANGe

*RST: INRange

24.17.7.4 Decode results

In all `BUS<m>:CSMB:RESult<n>` commands, the suffix <m> selects the serial bus, the suffix <n> selects the result number in the decode table, and the suffix <o> selects the cell number.

The following is an example for a set of query commands for bus 1 and result 1 is shown in the following, together with examples for outcomes of these queries:

- `:BUS1:CMSB:RCOunt?` !5
- `:BUS1:CMSB:RESult1:STATe?` !OK
- `:BUS1:CMSB:RESult1:STARt?` ! -0.0024964177
- `:BUS1:CMSB:RESult1:STOP?` ! -0.0024030384
- `:BUS1:CMSB:RESult1:TYPE?` !ff
- `:BUS1:CMSB:RESult1:CONE?` !0b11111111
- `:BUS1:CMSB:RESult1:CTWO?` !0xAA
- `:BUS1:CMSB:RESult1:CTHR?` !0xF590
- `:BUS1:CMSB:RESult1:CCOunt?` !5
- `:BUS1:CMSB:RESult1:CELL1:STAT?` !OK
- `:BUS1:CMSB:RESult1:CELL1:NAME?` !Start Delim
- `:BUS1:CMSB:RESult1:CELL1:VALue?` !101010101HL10HL0
- `:BUS1:CMSB:RESult1:CELL2:STAT?` !OK
- `:BUS1:CMSB:RESult1:CELL2:NAME?` !OP-FF
- `:BUS1:CMSB:RESult1:CELL2:VALue?` !0b11111111
- `:BUS1:CMSB:RESult1:CELL3:STAT?` !OK
- `:BUS1:CMSB:RESult1:CELL3:NAME?` !data
- `:BUS1:CMSB:RESult1:CELL3:VALue?` !0xAA
- `:BUS1:CMSB:RESult1:CELL4:STAT?` !OK
- `:BUS1:CMSB:RESult1:CELL4:NAME?` !CRC
- `:BUS1:CMSB:RESult1:CELL4:VALue?` !0xF590

- `:BUS1:CMSB:RESult1:CELL5:STAT?` !OK
- `:BUS1:CMSB:RESult1:CELL5:NAME?` !End Delim
- `:BUS1:CMSB:RESult1:CELL5:VALue?` !1HLHL101

To show the results on the screen, use the following commands:

- `BUS<m>:RESult` on page 1671

BUS<m>:CMSB:RCOunt? 1779
BUS<m>:CMSB:RESult<n>:STATe? 1779
BUS<m>:CMSB:RESult<n>:STARt? 1780
BUS<m>:CMSB:RESult<n>:STOP? 1780
BUS<m>:CMSB:RESult<n>:TYPE? 1781
BUS<m>:CMSB:RESult<n>:CONE? 1781
BUS<m>:CMSB:RESult<n>:CTWO? 1781
BUS<m>:CMSB:RESult<n>:CTHRee? 1781
BUS<m>:CMSB:RESult<n>:CCOunt? 1782
BUS<m>:CMSB:RESult<n>:CELL<o>:NAME? 1782
BUS<m>:CMSB:RESult<n>:CELL<o>:STARt? 1782
BUS<m>:CMSB:RESult<n>:CELL<o>:STOP? 1783
BUS<m>:CMSB:RESult<n>:CELL<o>:STATe? 1783
BUS<m>:CMSB:RESult<n>:CELL<o>:VALue? 1784

BUS<m>:CMSB:RCOunt?

Returns the count number of decoded result frames in a custom serial bus waveform. Basically, this is the maximum result index <n> when querying results by using `BUS<m>:CMSB:RESult<n>:XXX`.

Suffix:
<m> 1..4

Return values:
<Count>

Usage: Query only

BUS<m>:CMSB:RESult<n>:STATe?

Returns the overall state of the frame: either OK or the relevant error condition. R&S RTO6-K510 marks each frame with a status that indicates whether the decode succeeded or not.

Suffix:
<m> 1..4

<n> *

Return values:
<State> OK | LENGth | UNKNown | INComplete | CRC | PARity

OK
The frame was decoded normally and conforms to the frame description.

LENGth
The length error indicates that the frame ended prematurely, or an array in the frame had too few elements. The amount of bits that the software expected (based upon the user's frame description) was not found before the frame was terminated. This might occur because a new frame synchronized, or a gap appeared between the bits.

UNKNown
Unknown error

INComplete
The frame ended prematurely because it extends past the end of the record.

CRC
Checksum error in cyclic redundancy check (error in data)

PARity
Parity bit error, indicating a transmission error (only available if a parity is configured)

*RST: OK

Usage: Query only

BUS<m>:CMSB:RESult<n>:STARt?

Returns the start time of the frame.

Suffix:
<m> 1..4

<n> *

Return values:
<Start>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:CMSB:RESult<n>:STOP?

Returns the stop time of the frame.

Suffix:
<m> 1..4

<n> *

Return values:
<Stop>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:CMSB:RESult<n>:TYPE?

Returns the name of the selected frame (n) from the user defined frame format description, labeled according to `BUS<m>:CMSB:FRAMe<n>:TYPE` on page 1763.

Suffix:
<m> 1..4

<n> *

Return values:
<Type>

Usage: Query only

BUS<m>:CMSB:RESult<n>:CONE?

Returns the 1st cell content as specified in the "Result" column of the "Frame Format" specification table.

Suffix:
<m> 1..4

<n> *

Return values:
<Custom1>

Usage: Query only

BUS<m>:CMSB:RESult<n>:CTWO?

Returns the 2nd cell content as specified in the "Result" column of the "Frame Format" specification table.

Suffix:
<m> 1..4

<n> *

Return values:
<Custom2>

Usage: Query only

BUS<m>:CMSB:RESult<n>:CTHRee?

Returns the 3rd cell content as specified in the "Result" column of the "Frame Format" specification table.

Suffix:
<m> 1..4

<n> *

Return values:
<Custom3>

Usage: Query only

BUS<m>:CMSB:RESult<n>:CCOunt?

Returns the number of decoded cells in the specified result frame.

Suffix:
<m> 1..4

<n> *

Return values:
<NumWords> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:CMSB:RESult<n>:CELL<o>:NAME?

Returns the name of the specified cell. Cell names are not necessarily unique.

Suffix:
<m> 1..4

<n> *

<o> *

Return values:
<Name>

Usage: Query only

BUS<m>:CMSB:RESult<n>:CELL<o>:STARt?

Returns the start time of the selected cell.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *

<o> *
Selects the cell.

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:CMSB:RESult<n>:CELL<o>:STOP?

Returns the end time of the selected cell.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *

<o> *
Selects the cell.

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:CMSB:RESult<n>:CELL<o>:STATe?

Returns the state of the cell.

Suffix:

<m> 1..4

<n> *

<o> *

Return values:

<State> OK | LENGth | UNKNown | INComplete | CRC | PARity

OK
No error detected

LENGth
The length error indicates that the cell ended prematurely. The amount of bits that the software expected (based upon the user's frame description) was not found before the cell was terminated. This might occur because a new frame synchronized, or a gap appeared between the bits.

UNKNown
Unknown error

INComplete
The cell ended prematurely because it extends past the end of the record.

CRC
Checksum error in cyclic redundancy check (error in data)

PARity
Parity bit error, indicating a transmission error (only available if a parity is configured)

Usage: Query only

BUS<m>:CMSB:RESult<n>:CELL<o>:VALue?

Data content of the specified cell.

Suffix:

<m>	1..4
<n>	*
<o>	*

Return values:
<Value>

Usage: Query only

24.17.7.5 Search settings

SEARch:TRIGger:CMSB:ERENable 1785
SEARch:TRIGger:CMSB:ERRor<m>:ENABle 1785
SEARch:TRIGger:CMSB:FIENable 1785
SEARch:TRIGger:CMSB:FRENable 1785
SEARch:TRIGger:CMSB:BIT 1785
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:BIT 1785
SEARch:TRIGger:CMSB:DMAX 1786
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMAX 1786
SEARch:TRIGger:CMSB:DMIN 1786
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMIN 1786
SEARch:TRIGger:CMSB:DOPerator 1787
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DOPerator 1787
SEARch:TRIGger:CMSB:FRAMe<m>:ENABle 1787
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:ENABle 1787
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMAX 1788
SEARch:TRIGger:CMSB:IMAX 1788
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMIN 1788
SEARch:TRIGger:CMSB:IMIN 1788
SEARch:TRIGger:CMSB:IOPerator 1788
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IOPerator 1788

SEARch:TRIGger:CMSB:ERENable <SearchName>,<ErrorName>, <Enabler>
SEARch:TRIGger:CMSB:ERRor<m>:ENABle <SearchName>,<Enable>
SEARch:TRIGger:CMSB:ERRor<m>:ENABle? <SearchName>

Defines the error type to be searched for. You can search for all error types in parallel.

Suffix:
<m> *

Parameters:
<Enable> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CMSB:FIENable <SearchName>,<Frame>, <Field>, <Enabler>

Enables or disables the specific field within the defined frame to be searched for.

Setting parameters:
<SearchName>

<Frame>

<Field>

<Enabler> ON | OFF

Usage: Setting only

SEARch:TRIGger:CMSB:FRENable <SearchName>,<Frame>, <Enabler>

Enables or disables the specific frame to be searched for.

Setting parameters:
<SearchName>

<Frame>

<Enabler> ON | OFF

Usage: Setting only

SEARch:TRIGger:CMSB:BIT <SearchName>,<Frame>, <Field>, <Bit>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:BIT <SearchName>,<BitState>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:BIT? <SearchName>

Sets the bit state of a field to be searched that only consists of one bit.

Suffix:
<m> *

<n> *

Parameters:

<BitState> ONE | ZERO | DC

ONE
1

ZERO
0

DC
"Do not care" (DC) = X

*RST: DC

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CMSB:DMAX <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMAX <SearchName>,<DataMax>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMAX? <SearchName>

Sets the end value of a data pattern range if `SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DOPerator` is set to `INRange` or `OORange`.

Suffix:

<m> *

<n> *

Parameters:

<DataMax> Specifies the name of the generic trigger setting frame.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CMSB:DMIN <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMIN <SearchName>,<DataMin>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMIN? <SearchName>

Specifies the data pattern to be searched, or sets the start value of a data pattern range to be searched.

Suffix:

<m> *

<n> *

Parameters:

<DataMin> Specifies the name of the generic trigger setting frame.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CMSB:DOPerator <SearchName>,<Frame>, <Field>, <Operator>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DOPerator
<SearchName>,<DataOperator>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DOPerator? <SearchName>

Sets the operator to set a specific data pattern to be searched in the selected field of the selected frame.

Suffix:
<m> *

<n> *

Parameters:
<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one endpoint value to be set using `SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMIN`.

INRange | OORange
In range, out of range. These conditions require a range of endpoint values to be set using `SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMIN` and `SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CMSB:FRAMe<m>:ENABle <SearchName>,<Enable>
SEARch:TRIGger:CMSB:FRAMe<m>:ENABle? <SearchName>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:ENABle
<SearchName>,<CondEnabler>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:ENABle? <SearchName>

Enables or disables the checking condition for searching a specific data pattern in the selected field of the selected frame.

Suffix:
<m> *

<n> *

Parameters:
<CondEnabler> ON | OFF

ON
Checking condition enabled

OFF
Checking condition disabled

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMAX <SearchName>,<IndexMax>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMAX? <SearchName>
SEARch:TRIGger:CMSB:IMAX <SearchName>,<Frame>, <Field>, <Data>

Sets the end value of the bit index range (data position range), if `SEARch:TRIGger:CMSB:IOPerator` is set to `INRange`.

Setting parameters:
<SearchName>

<Frame>

<Field>

<Data>

Usage: Setting only

SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMIN <SearchName>,<IndexMin>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMIN? <SearchName>
SEARch:TRIGger:CMSB:IMIN <SearchName>,<Frame>, <Field>, <Data>

Sets the bit index (data position).

Setting parameters:
<SearchName>

<Frame>

<Field>

<Data>

Usage: Setting only

SEARch:TRIGger:CMSB:IOPerator <SearchName>,<Frame>, <Field>, <Operator>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IOPerator
<SearchName>,<IndexOperator>
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IOPerator? <SearchName>

Sets the operator for the index for searching in the selected field of the selected frame.

Suffix:
<m> *

<n> *

Parameters:
<IndexOperator> EQUal | INRange | RANGe

EQUal
This condition requires one endpoint value to be set using `SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMIN` and `SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMAX`.

*RST: INRange

Parameters for setting and query:
<SearchName>

24.17.7.6 Search results

SEARch:RESult:CMSB<m>:FCOunt? 1789
SEARch:RESult:CMSB<m>:FRAMe<n>:CCOunt? 1789
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:NAME? 1790
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STARt? 1790
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STATe? 1791
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STOP? 1791
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:VALue? 1792
SEARch:RESult:CMSB<m>:FRAMe<n>:CONE? 1792
SEARch:RESult:CMSB<m>:FRAMe<n>:CTHRee? 1792
SEARch:RESult:CMSB<m>:FRAMe<n>:CTWO? 1793
SEARch:RESult:CMSB<m>:FRAMe<n>:STARt? 1793
SEARch:RESult:CMSB<m>:FRAMe<n>:STATe? 1793
SEARch:RESult:CMSB<m>:FRAMe<n>:STOP? 1794
SEARch:RESult:CMSB<m>:FRAMe<n>:TYPE? 1794

SEARch:RESult:CMSB<m>:FCOunt? <SearchName>

Returns the count number of decoded result frames in a custom serial bus waveform. Basically, this is the maximum result index <n> when querying results by using `SEARch:RESult:CMSB<m>:FRAMe<n>:XXX`.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CCOunt? <SearchName>

Returns the count number of cell in the frames, within the search result, in a custom serial bus waveform.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<NumWords> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:NAME? <SearchName>

Returns the name of the specified cell within the search result. Cell names are not necessarily unique.

Suffix:
<m> *

<n> *

<o> *

Query parameters:
<SearchName>

Return values:
<Name>

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STARt? <SearchName>

Returns the start time of the cell within the search result.

Suffix:
<m> *

<n> *

<o> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STATe? <SearchName>

Returns the state of the cell within the search result.

Suffix:

<m> *

<n> *

<o> *

Query parameters:

<SearchName>

Return values:

<State> OK | LENGth | UNKNown | INComplete | CRC | PARity

OK
No error detected

LENGth
The length error indicates that the cell ended prematurely. The amount of bits that the software expected (based upon the user's frame description) was not found before the cell was terminated. This might occur because a new frame synchronized, or a gap appeared between the bits.

UNKNown
Unknown error

INComplete
The cell ended prematurely because it extends past the end of the record.

CRC
Checksum error in cyclic redundancy check (error in data)

PARity
Parity bit error, indicating a transmission error (only available if a parity is configured)

*RST: OK

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STOP? <SearchName>

Returns the stop time of the cell within the search result.

Suffix:

<m> *

<n> *

<o> *

Query parameters:

<SearchName>

Return values:

<Stop>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:VALue? <SearchName>

Data content of the specified cell within the search result.

Suffix:

<m> *

<n> *

<o> *

Query parameters:

<SearchName>

Return values:

<Value>

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CONE? <SearchName>

Returns the 1st field content within the search result.

Suffix:

<m> *

<n> *

Query parameters:

<SearchName>

Return values:

<Custom1>

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CTHRee? <SearchName>

Returns the 3rd field content.

Suffix:

<m> *

<n> *

Query parameters:

<SearchName>

Return values:
<Custom3>

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:CTWO? <SearchName>

Returns the 2nd field content.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Custom2>

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:STARt? <SearchName>

Returns the start time of the frame within the search result.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Start>

Range:	-100E+24 to 100E+24
Increment:	100E-12
*RST:	0
Default unit:	s

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:STATe? <SearchName>

Returns the overall state of the frame within the search result: either OK or the relevant error condition. R&S RTO6-K510 marks each frame with a status that indicates whether the decode succeeded or not.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<State> OK | LENGth | UNKNown | INComplete | CRC | PARity

OK
The frame was decoded normally and conforms to the frame description.

LENGth
The length error indicates that the frame ended prematurely, or an array in the frame had too few elements. The amount of bits that the software expected (based upon the user's frame description) was not found before the frame was terminated. This might occur because a new frame synchronized, or a gap appeared between the bits.

UNKNown
Unknown error

INComplete
The frame ended prematurely because it extends past the end of the record.

CRC
Checksum error in cyclic redundancy check (error in data)

*RST: OK

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:STOP? <SearchName>

Returns the stop time of the frame within the search result.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Stop>

Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:CMSB<m>:FRAMe<n>:TYPE? <SearchName>

Returns the name of the selected frame (n) from the user defined frame format description, labeled according to `BUS<m>:CMSB:FRAMe<n>:TYPE` on page 1763.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Type>

Usage: Query only

24.17.8 CAN (option R&S RTO6-K520)

- Configuration....1795
- Trigger....1800
- Decode results....1808
- Search settings....1817
- Search results....1825
- Symbolic trigger, decode and search....1832

24.17.8.1 Configuration

BUS<m>:CAN:DATA:SOURce....1795
BUS<m>:CAN:TYPE....1796
BUS<m>:CAN:FDATa:PSTandard....1796
BUS<m>:CAN:DATA:THReshold....1796
BUS<m>:CAN:TECHnology....1797
BUS<m>:CAN:BITRate....1797
BUS<m>:CAN:FDATa:ENABle....1798
BUS<m>:CAN:FDATa:DBITrate....1798
BUS<m>:CAN:FDATa:SAMPlepoint....1798
BUS<m>:CAN:SAMPlepoint....1798
BUS<m>:CAN:FDATa:T1Segment....1799
BUS<m>:CAN:T1Segment....1799
BUS<m>:CAN:FDATa:T2Segment....1799
BUS<m>:CAN:T2Segment....1799
BUS<m>:CAN:FDATa:JWIDth....1800
BUS<m>:CAN:JWIDth....1800

BUS<m>:CAN:DATA:SOURce <DataSource>

Sets the source of the data line that is selected with `BUS<m>:CAN:TYPE`.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<DataSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

*RST: C1W1

Usage: Asynchronous command

BUS<m>:CAN:TYPE <SignalType>

Selects the CAN-High or CAN-Low line. Both lines are required for differential signal transmission used by CAN.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<SignalType> CANH | CANL

*RST: CANL

BUS<m>:CAN:FDATa:PSTandard <ProtStd>

Only available for CAN FD buses. Selects whether the tested signal is an ISO CAN FD signal or not.

Suffix:
<m> 1..4

Parameters:
<ProtStd> ISO | NISO

ISO
Signals are decoded according to the the ISO CAN FD protocol. This protocol has an additional stuff count field before the CRC sequence.

NISO
Non-ISO. Signals are decoded according to the the Bosch CAN FD protocol.

*RST: ISO

Firmware/software: FW 3.35

BUS<m>:CAN:DATA:THReshold <Threshold>

Sets a user-defined threshold value.

Alternatively, you can set the threshold according to the signal technology with `BUS<m>:CAN:TECHnology`.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Threshold> Range: -12 to 12
Increment: 0.1
*RST: 0
Default unit: V

BUS<m>:CAN:TECHnology <Technology>

Sets the threshold voltage as defined for various signal technologies.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Technology> V25 | V3 | V2 | V0 | MANual

V25
2.5 Volt (CMOS 5.0 V)

V3
3.0 Volt (CAN_H HS / CAN_L LS)

V2
2.0 Volt (CAN_L HS / CAN_H LS)

V0
Ground

MANual
Manual setting of user-defined values with `BUS<m>:CAN:DATA:THReshold`.

*RST: V25

BUS<m>:CAN:BITRate <Bitrate>

For CAN buses, the "Bit rate" sets the number of transmitted bits per second.

For CAN FD buses, this parameter is called "Arbitration rate" and sets the bit rate of the arbitration phase.

The maximum bit rate for High Speed CAN is 1 Mbit/s. The bit rate is uniform and fixed for a given CAN or CAN FD bus.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:

<Bitrate>	Range:	100 to 5E+6
	Increment:	1
	*RST:	100E+3
	Default unit:	bps

BUS<m>:CAN:FDATa:ENABle <CANFDEnabled>

Enables the CAN FD protocol configuration.

The setting is available in CAN FD option.

Suffix:

<m>	1..4

Parameters:

<CANFDEnabled>	ON | OFF	
	*RST:	ON

BUS<m>:CAN:FDATa:DBITrate <FlexDatBitrate>

Sets the bit rate of the data phase. The data rate can be higher than the arbitration rate, but it is uniform and fixed for a given CAN FD bus.

The setting is available in CAN FD option.

Suffix:

<m>	1..4

Parameters:

<FlexDatBitrate>	Range:	100 to 15E+6
	Increment:	1
	*RST:	1E+6
	Default unit:	bps

BUS<m>:CAN:FDATa:SAMPlepoint <FlexDatSmpPt>
BUS<m>:CAN:SAMPlepoint <SamplePoint>

Sets the position of the sample point within the bit in percent of the nominal bit time.

Alternatively, you can set the sample point with `BUS<m>:CAN:T1Segment` and `BUS<m>:CAN:T2Segment`.

For CAN FD signals, `BUS<m>:CAN:SAMPlepoint` defines the synchronization of the arbitration phase, and `BUS<m>:CAN:FDATa:SAMPlepoint` defines the synchronization of the data phase.

Suffix:

<m>	1..4 Selects the serial bus.

Parameters:

<FlexDatSmpPt> <SamplePoint>	Range:	12 to 96 if number of segments is 24. The range reduces if the signal has fewer segments.
	Increment:	5
	*RST:	66
	Default unit:	%

BUS<m>:CAN:FDATa:T1Segment <FlexDatTimeSeg1>
BUS<m>:CAN:T1Segment <TimeSeg1>

Sets the number of time quanta before the sample point (T1Segment). T1Segment comprises the segments Synch_seg, Prop_seg, and Phase_seg1 which are specified in the CAN standard.

Make sure to set also `BUS<m>:CAN:T2Segment` for correct definition of the sample point. Alternatively, you can use `BUS<m>:CAN:SAMPlepoint`.

See also: Chapter 13.7.2.2, "Advanced settings", on page 582

For CAN FD signals, `BUS<m>:CAN:T1Segment` defines the synchronization of the arbitration phase, and `BUS<m>:CAN:FDATa:T1Segment` defines the synchronization of the data phase.

Suffix:

<m>	1..4

Parameters:

<FlexDatTimeSeg1> <TimeSeg1>	Time quanta	
	Range:	3 to 23
	Increment:	1
	*RST:	6.6

BUS<m>:CAN:FDATa:T2Segment <FlexDatTimeSeg2>
BUS<m>:CAN:T2Segment <TimeSeg2>

Sets the number of time quanta after the sample point (T2Segment). T2Segment matches Phase_seg2 specified in the CAN standard.

Make sure to set also `BUS<m>:CAN:T1Segment` on page 1799 for correct definition of the sample point. Alternatively, you can use `BUS<m>:CAN:SAMPlepoint`.

See also: Chapter 13.7.2.2, "Advanced settings", on page 582

For CAN FD signals, `BUS<m>:CAN:T2Segment` defines the synchronization of the arbitration phase, and `BUS<m>:CAN:FDATa:T2Segment` defines the synchronization of the data phase.

Suffix:

<m>	1..4 Selects the serial bus.

Parameters:

<FlexDatTimeSeg2>
<TimeSeg2>

Time quanta

Range: 1 to 21
Increment: 1
*RST: 3.4

BUS<m>:CAN:FDATa:JWIDth <FlexDatJumpWdt>
BUS<m>:CAN:JWIDth <JumpWidth>

Defines the maximum number of time quanta for phase correction. Time segment1 may be lengthened or Time segment2 may be shortened due to resynchronization. Resynchronization corrects the phase error of an edge caused by the drift of the oscillators.

For CAN FD signals, this setting defines the synchronization of the arbitration phase.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<JumpWidth>

Time quanta

Range: 1 to 4, available maximum depends on the number of segments and the sample point
Increment: 1
*RST: 1

24.17.8.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:CAN:TYPE 1801
TRIGger<m>:CAN:FDATa:STANdard 1802
TRIGger<m>:CAN:FTYPe 1802
TRIGger<m>:CAN:ITYPe 1802
TRIGger<m>:CAN:ICONdition 1803
TRIGger<m>:CAN:IMIN 1803
TRIGger<m>:CAN:IMAX 1803
TRIGger<m>:CAN:FDATa:FDF 1803
TRIGger<m>:CAN:FDATa:BRS 1804
TRIGger<m>:CAN:FDATa:ESI 1804
TRIGger<m>:CAN:DCONdition 1804
TRIGger<m>:CAN:DMIN 1804

TRIGger<m>:CAN:DMAX 1805
TRIGger<m>:CAN:BORDer 1805
TRIGger<m>:CAN:DLCCondition 1805
TRIGger<m>:CAN:DLC 1805
TRIGger<m>:CAN:NDBYtes? 1806
TRIGger<m>:CAN:FDATa:DPOPerator 1806
TRIGger<m>:CAN:FDATa:DPOSition 1806
TRIGger<m>:CAN:FDATa:DPTO 1807
TRIGger<m>:CAN:ACKerror 1807
TRIGger<m>:CAN:BITSterror 1807
TRIGger<m>:CAN:CRCerror 1807
TRIGger<m>:CAN:FORMerror 1807
TRIGger<m>:CAN:FDATa:SCERror 1808

TRIGger<m>:CAN:TYPE <Type>

Selects the trigger type for CAN analysis.

See: "Trigger type" on page 587.

Parameters:

<Type> STOF | FTYP | ID | IDDT | ERRC

STOF
STart Of Frame: triggers on the first edge of the dominant SOF bit (synchronization bit).

FTYP
Frame TYPe: triggers on a specified frame type (data, remote, error, or overload) and on the identifier format.
To set the frame type, use `TRIGger<m>:CAN:FTYPe`. ´Set the identifier format with `TRIGger<m>:CAN:ITYPe`

ID
IDentifier: Sets the trigger to one specific identifier or an identifier range. To set the identifier, use `TRIGger<m>:CAN:ICONdition`, `TRIGger<m>:CAN:IMIN`, and `TRIGger<m>:CAN:IMAX`.

IDDT
IDentifier and DaTa: Combination of identifier and data conditions To set the identifier condition, use `TRIGger<m>:CAN:ICONdition`, `TRIGger<m>:CAN:IMIN`, and `TRIGger<m>:CAN:IMAX`.
To set the data condition, use `TRIGger<m>:CAN:DCONdition`, `TRIGger<m>:CAN:DMIN`, and `TRIGger<m>:CAN:DMAX`.

ERRC
ERRor Condition: Define the error types with
`TRIGger<m>:CAN:ACKerror`,
`TRIGger<m>:CAN:BITSterror`,
`TRIGger<m>:CAN:CRCerror`,
`TRIGger<m>:CAN:FORMerror`,
`TRIGger<m>:CAN:FDATa:SCERror` on page 1808.

*RST: STOF

TRIGger<m>:CAN:FDATa:STANdard <Standard>

Selects the CAN standard. Use `ANY` if the standard of the signal is unknown.

The setting is available in CAN FD option.

Parameters:

<Standard> ANY | CAN | CANFd

*RST: CAN

TRIGger<m>:CAN:FTYPe <FrameType>

Selects the CAN frame type if `TRIGger<m>:CAN:TYPE` is set to FTYP (frame type) or ID (identifier).

For data and remote frames, the identifier format has to be set with `TRIGger<m>:CAN:ITYPe`.

See also: "Frame type" on page 589

Parameters:

<FrameType> ANY | DATA | REMote | ERRor | OVERload

Available values depend on the CAN standard and on the `TRIGger<m>:CAN:TYPE` setting:
Remote frames are not available in the CAN FD protocol.
If the trigger type is set to FTYP (frame type), you can set the values `DATA | REMote | ERRor | OVERload`.
If the trigger type is set to ID (identifier), you can set the values `ANY | DATA | REMote`.

*RST: ANY

TRIGger<m>:CAN:ITYPe <IdentifierType>

Selects the format of data and remote frames.

Remote frames are not available in the CAN FD protocol.

Parameters:

<IdentifierType> ANY | B11 | B29

B11
11 bit identifier (standard format). The instrument triggers on the sample point of the IDE bit.

B29
29 bit identifier (extended format). The instrument triggers on the sample point of the RTR bit.

ANY
The ID type and ID pattern are not relevant for the trigger condition.

*RST: ANY

TRIGger<m>:CAN:ICONdition <IdOperator>

Sets the operator to set a specific identifier or an identifier range.

Parameters:

<IdOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These condition require one identifier pattern to be set with `TRIGger<m>:CAN:IMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:CAN:IMIN` and `TRIGger<m>:CAN:IMAX` on page 1803.

*RST: EQUal

TRIGger<m>:CAN:IMIN <IdPattern>

Specifies a message identifier pattern, or sets the the start value of an identifier range.

Parameters:

<IdPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

TRIGger<m>:CAN:IMAX <IdPatternTo>

Sets the the end value of an identifier range if `TRIGger<m>:CAN:ICONdition` is set to `INRange` or `OORange`.

Parameters:

<IdPatternTo> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

TRIGger<m>:CAN:FDATa:FDF <FDFBit>

Specifies the CAN FD frame format. It corresponds to the EDL bit (extended data length), which only exists in CAN FD format.

Parameters:

<FDFBit> ONE | ZERO | DC

ONE: CAN FD.
ZERO: CAN.
DC: don't care, the format is not relevant.

*RST: DC

TRIGger<m>:CAN:FDATa:BRS <BRSBit>

Sets the bit rate switch bit.

Parameters:

<BRSBit> ONE | ZERO | DC

ONE: the bit rate switches from the bit rate of the arbitration phase to the faster data rate.

*RST: ONE

TRIGger<m>:CAN:FDATa:ESI <ESIBit>

Sets the error state indicator bit.

Parameters:

<ESIBit> ONE | ZERO | DC

DC: don't care, bit is nor relevant

*RST: DC

TRIGger<m>:CAN:DCONdition <DataOperator>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with `TRIGger<m>:CAN:DMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:CAN:DMIN` and `TRIGger<m>:CAN:DMAX`.

*RST: EQUal

TRIGger<m>:CAN:DMIN <DataPattern>

Specifies a data pattern, or sets the the start value of a data pattern range.

Parameters:

<DataPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:CAN:DMAX <DataPatternTo>

Sets the the end value of an data range if `TRIGger<m>:CAN:DCONdition` is set to `INRange` or `OORange`.

Parameters:

<DataPatternTo> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:CAN:BORDer <Endianness>

Sets the byte order (endianness) of the data transfer. Only for CAN protocol.

Parameters:

<Endianness> BENDian | LENDian

BENDian
Big endian, data is analyzed and evaluated in the order of reception.

LENDian
Little endian, the instrument reads the complete data, reverses the byte order of the data, and compares it with the specified data word.

*RST: BENDian

TRIGger<m>:CAN:DLCCondition <DLCOperator>

Operator to set the data length code for triggering on CAN and CAN FD data.

For details, see "Data setup: DLC, NDB, Transfer, Condition, Data min, Data max" on page 591.

The number of data bytes to be found is set with `TRIGger<m>:CAN:DLC`.

See also: `TRIGger<m>:CAN:BORDer`.

Parameters:

<DLCOperator> EQUal | GETHan

For little endian transfer direction, EQUal must be set.

*RST: GETHan

TRIGger<m>:CAN:DLC <WordCount>

Sets the Data Length Code, the number of data bytes to be found. For complete definition, set also the operator with `TRIGger<m>:CAN:DLCCondition` on page 1805.

Parameters:

<WordCount> Range: CAN: 1 to 8, CAN FD: 1 to 15 (64 bytes)
Increment: 1
*RST: 1

TRIGger<m>:CAN:NDBYtes?

Returns the number of data bytes defined by DLC. DLC and NDB are different in CAN FD for DLCs > 8.

See also: "Data setup: DLC, NDB, Transfer, Condition, Data min, Data max" on page 591.

Return values:

<NDBytes> Range: 1 to 64
Increment: 1
*RST: 1

Usage: Query only

TRIGger<m>:CAN:FDATa:DPOPerator <DataPosOperator>

Sets the operator to define an exact position or a data range where the instrument looks for the specified data pattern.

The setting is available in CAN FD option.

The position can be defined if the data field of the frame is longer than 8 bytes - if `TRIGger<m>:CAN:DLC`≥9.

Parameters:

<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe

ANY = OFF
The data position is not relevant for the trigger condition.

EQUal | GETHan
Equal, Greater or equal than. These conditions require one data position to be set with `TRIGger<m>:CAN:FDATa:DPOSition`.

INRange = RANGe
In range: Set the minimum and maximum value of the range with `TRIGger<m>:CAN:FDATa:DPOSition` and `TRIGger<m>:CAN:FDATa:DPTO`.

*RST: ANY

TRIGger<m>:CAN:FDATa:DPOSition <DataPosition>

Defines the number of the first data byte at which the data pattern may start.

The setting is available in CAN FD option.

Parameters:

<DataPosition> Range: 1 to 57
Increment: 1
*RST: 1

TRIGger<m>:CAN:FDATa:DPTO <DataPositionTo>

Sets the number of the last byte at which the required data pattern may start.

Parameters:

<DataPositionTo>	Range:	8 to 64
	Increment:	1
	*RST:	8

TRIGger<m>:CAN:ACKerror <AckError>

Triggers when the transmitter does not receive an acknowledgment - a dominant bit during the Ack Slot.

The trigger type has to be set before: `TRIGger<m>:CAN:TYPE` to `ERRC`.

Parameters:

<AckError>	ON | OFF	
	*RST:	ON

TRIGger<m>:CAN:BITSterror <BitStuffError>

Triggers if astuff error occurs - when the 6th consecutive equal bit level in the mentioned fields is detected.

The trigger type has to be set before: `TRIGger<m>:CAN:TYPE` to `ERRC`.

Parameters:

<BitStuffError>	ON | OFF	
	*RST:	ON

TRIGger<m>:CAN:CRCerror <ChecksumError>

Triggers on CRC errors. A CRC error occurs when the CRC calculated by the receiver differs from the received value in the CRC sequence.

The trigger type has to be set before: `TRIGger<m>:CAN:TYPE` to `ERRC`.

Parameters:

<ChecksumError>	ON | OFF	
	*RST:	ON

TRIGger<m>:CAN:FORMerror <FormError>

Triggers when a fixed-form bit field contains one or more illegal bits.

The trigger type has to be set before: `TRIGger<m>:CAN:TYPE` to `ERRC`.

Parameters:

<FormError>	ON | OFF	
	*RST:	ON

TRIGger<m>:CAN:FDATa:SCERror <StuffCountError>

Triggers on stuff count errors. A stuff count error occurs if th received stuff count value does not match the value calculated from the own stuff bit count.

The trigger type `TRIGger<m>:CAN:TYPE` must be set to `ERRC`.

Only relevant for CAN FD signals in ISO standard.

Parameters:

<StuffCountError> ON | OFF

*RST: ON

24.17.8.3 Decode results

To load and acivate a label list, use:

- `BUS<m>:NEWList` on page 1672
- `BUS<m>:SYMBols` on page 1672

BUS<m>:CAN:FCOunt? 1808
BUS<m>:CAN:FDATa:FRAMe<n>:STANdard? 1809
BUS<m>:CAN:FRAMe<n>:STATus? 1809
BUS<m>:CAN:FRAMe<n>:NDBYtes? 1810
BUS<m>:CAN:FRAMe<n>:STARt? 1810
BUS<m>:CAN:FRAMe<n>:STOP? 1810
BUS<m>:CAN:FRAMe<n>:SYMBol? 1811
BUS<m>:CAN:FRAMe<n>:TYPE? 1811
BUS<m>:CAN:FRAMe<n>:DATA? 1811
BUS<m>:CAN:FRAMe<n>:ACKState? 1812
BUS<m>:CAN:FRAMe<n>:CSSTate? 1812
BUS<m>:CAN:FRAMe<n>:DLCState? 1812
BUS<m>:CAN:FRAMe<n>:IDSTate? 1812
BUS<m>:CAN:FRAMe<n>:ACKValue? 1812
BUS<m>:CAN:FRAMe<n>:BITRate? 1813
BUS<m>:CAN:FRAMe<n>:CSValue? 1813
BUS<m>:CAN:FRAMe<n>:DLCValue? 1813
BUS<m>:CAN:FRAMe<n>:IDTYpe? 1814
BUS<m>:CAN:FRAMe<n>:IDValue? 1814
BUS<m>:CAN:FRAMe<n>:BSEPosition? 1815
BUS<m>:CAN:FRAMe<n>:FERCause? 1815
BUS<m>:CAN:FRAMe<n>:SDEXport? 1815
BUS<m>:CAN:FDATa:FRAMe<n>:SCValue? 1816
BUS<m>:CAN:FRAMe<n>:BYTE<o>:STATe? 1816
BUS<m>:CAN:FRAMe<n>:BYTE<o>:VALue? 1816

BUS<m>:CAN:FCOunt?

Returns the number of decoded frames of the acquisition.

Suffix:
<m> 1..4
Selects the serial bus.

Return values:
<Count> Total number of decoded frames.

Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:CAN:FDATa:FRAMe<n>:STANdard?

Returns the CAN standard.

The setting is available in CAN FD option.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
The frame suffix is irrelevant.

Return values:
<Standard> CAN | CANFd

*RST: CAN

Usage: Query only

BUS<m>:CAN:FRAMe<n>:STATus?

Returns the overall state of the selected frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<FrameState> OK | FORM | BTST | CRC | CRCD | NOACk | ACKD | EOFD | CAERror | FCERor | INSufficient | SERRror | SFERror | SCERror | SAERror | SCAE | SCFE

OK: the frame is valid.
FORM: Fixed-bit form error
BTST: Bit stuffing error occured.
CRC: Cyclic redundancy check failed.
CRCD: Wrong CRC delimiter occured.
NOACK: Acknowlegde is missing.
ACKD: Wrong ACK delimiter occured.

EOFD: Wrong end of frame.
CAERror: CRC error followed by an acknowledgement error (missing acknowledge)
FCERor: CRC error followed by a form error (wrong CRC delimiter or wrong ACK delimiter)
INSufficient: The frame is not completely contained in the acquisition. The acquired part of the frame is valid.
SERRror: Stuff count error (CAN-FD ISO only)
SFER: Stuff count error and FORM error (CAN-FD ISO only)
SCER: Stuff count error and CRC error (CAN-FD ISO only)
SAER: Stuff count error and ACK error (CAN-FD ISO only)
SCAE: Stuff count error and CRC error and ACK error (CAN-FD ISO only)
SCFE: Stuff count error and CRC error and FORM error (CAN-FD ISO only)

*RST: OK

Usage: Query only

BUS<m>:CAN:FRAMe<n>:NDBYtes?

REturns the number of data bytes.

Suffix:

<m> 1..4

<n> *

Return values:

<NDBytes> Range: 1 to 64
Increment: 1
*RST: 1

Usage: Query only

Firmware/software: FW 3.35

BUS<m>:CAN:FRAMe<n>:STARt?
BUS<m>:CAN:FRAMe<n>:STOP?

Return the start time and stop time of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Start>, <Stop> Time

Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:CAN:FRAMe<n>:SYMBol?

Returns the symbolic label of the specified frame if the label list is enabled.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the number of the frame in the current acquisition, 1...n.

Return values:

<Label> String with symbolic label of the identifier

Example: `BUS:CAN:FRAMe:SYMBol?`
Response: `Temperature`

Usage: Query only

BUS<m>:CAN:FRAMe<n>:TYPE?

Returns the frame type of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameType> DATA | REMote | ERR | OVLD

Data, remote, error or overload frame.

*RST: DATA

Usage: Query only

BUS<m>:CAN:FRAMe<n>:DATA?

Returns the data of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<Data> Comma-separated list of values. The first value is the number of bytes, followed by the values of the data bytes.

Example:
```
BUS1:CAN:FRAMe2:DATA?
--> 3,208,231,32
```
Returns the data of the second frame: the number of bytes is 3 data (first value).

Usage: Query only

BUS<m>:CAN:FRAMe<n>:ACKState?
BUS<m>:CAN:FRAMe<n>:CSSTate?
BUS<m>:CAN:FRAMe<n>:DLCState?
BUS<m>:CAN:FRAMe<n>:IDSTate?

Return the states of following parts of a message

- ACKState: state of acknowledgement field
- CSSTate: state of checksum field (CRC)
- DLCState: state of data length code
- IDSTate: identifier state

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<State> OK | ERRor | UNDF
UNDF: Undefined
*RST: OK

Usage: Query only

BUS<m>:CAN:FRAMe<n>:ACKValue?

Returns the value of the acknowledge slot for the selected frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<AckValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: 0 to 1
*RST: 0

Usage: Query only

BUS<m>:CAN:FRAMe<n>:BITRate?

Returns the primary bit rate.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<PrimaryBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

BUS<m>:CAN:FRAMe<n>:CSValue?

Returns the CRC sequence value of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<ChecksumValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: 0 to 2097151
*RST: 0

Usage: Query only

BUS<m>:CAN:FRAMe<n>:DLCValue?

Returns the data length code of the selected frame - the number of data bytes in the frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameDLCValue> Number of data bytes in decimal values.

Range: 0 to 15
*RST: 0

Usage: Query only

BUS<m>:CAN:FRAMe<n>:IDTYpe?

Returns the identifier type of the selected frame, the identifier format of data and remote frames.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<IdentifierType> ANY | B11 | B29

B11: standard format, 11 bit
B29: extended format, 29 bit

*RST: B11

Usage: Query only

BUS<m>:CAN:FRAMe<n>:IDValue?

Returns the identifier value of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<IdentifierValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: 0 to 536870911
*RST: 0

Usage: Query only

BUS<m>:CAN:FRAMe<n>:BSEPosition?

Returns the location of a bit stuffing error.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<BitStuffErrorPos> Time when the error occured

Range: 0 to 5000
*RST: 0

Usage: Query only

BUS<m>:CAN:FRAMe<n>:FERCause?

Returns information on a form error, if the frame status query returned a FORM error.

See also: BUS<m>:CAN:FRAMe<n>:STATus? on page 1809

Suffix:

<m> 1..4

<n> *

Return values:

<FormErrorCause> NONE | CRCDerror | ACKDerror | FSBE | RESerror

CRCD = CRC delimiter error
ACKD = ACK delimiter error
FSBE = Fixed stuff bit error (CAN-FD ISO only)
RESE = Reserved bit error

*RST: NONE

Usage: Query only

BUS<m>:CAN:FRAMe<n>:SDEXport?

Returns the symbolic data of the frame in export format.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<SymbolicData>

Usage: Query only

BUS<m>:CAN:FDATa:FRAMe<n>:SCValue?

Returns the stuff bit count modulo 8 value.

Suffix:

<m>	1..4
<n>	*

Return values:

<StuffCount>	Range:	0 to 7
	Increment:	1
	*RST:	0

Usage: Query only

BUS<m>:CAN:FRAMe<n>:BYTE<o>:STATe?

Returns the state of the specified byte.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.
<o>	* Selects the byte number.

Return values:

<State>	OK | ERRor | UNDF UNDF: Undefined *RST: OK

Usage: Query only

BUS<m>:CAN:FRAMe<n>:BYTE<o>:VALue?

Returns the value of the specified byte.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.
<o>	* Selects the byte number.

Return values:

<Value> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: 0 to 255
*RST: 0

Usage: Query only

24.17.8.4 Search settings

SEARch:TRIGger:CAN[:SSOFrame] 1817
SEARch:TRIGger:CAN:SFTYpe 1818
SEARch:TRIGger:CAN:SFIDentifier 1818
SEARch:TRIGger:CAN:SIDData 1818
SEARch:TRIGger:CAN:SERRor 1818
SEARch:TRIGger:CAN:FDATa:STANdard 1819
SEARch:TRIGger:CAN:FTYPe 1819
SEARch:TRIGger:CAN:ITYPe 1819
SEARch:TRIGger:CAN:ICONdition 1819
SEARch:TRIGger:CAN:IMAX 1820
SEARch:TRIGger:CAN:IMIN 1820
SEARch:TRIGger:CAN:DCONdition 1820
SEARch:TRIGger:CAN:DMIN 1821
SEARch:TRIGger:CAN:DMAX 1821
SEARch:TRIGger:CAN:DLCCondition 1821
SEARch:TRIGger:CAN:DLC 1822
SEARch:RESult:CAN:FRAMe<m>:NDBYtes? 1822
SEARch:TRIGger:CAN:FDATa:DPOPerator 1822
SEARch:TRIGger:CAN:FDATa:DPOSition 1823
SEARch:TRIGger:CAN:FDATa:DPTO 1823
SEARch:TRIGger:CAN:ACKerror 1823
SEARch:TRIGger:CAN:BITSterror 1824
SEARch:TRIGger:CAN:CRCerror 1824
SEARch:TRIGger:CAN:FORMerror 1824
SEARch:TRIGger:CAN:FDATa:SCERror 1824
SEARch:TRIGger:CAN:FDATa[:FDF] 1825
SEARch:TRIGger:CAN:FDATa:BRS 1825
SEARch:TRIGger:CAN:FDATa:ESI 1825

SEARch:TRIGger:CAN[:SSOFrame] <SearchName>,<FrameStart>
SEARch:TRIGger:CAN[:SSOFrame]? <SearchName>

Enables the search for a start of frame.

Parameters:

<FrameStart> ON | OFF

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:SFTYpe <SearchName>,<FrameType>
SEARch:TRIGger:CAN:SFTYpe? <SearchName>

Enables the search for a specified frame type.

Parameters:
<FrameType> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:SFIDentifier <SearchName>,<Identifier>
SEARch:TRIGger:CAN:SFIDentifier? <SearchName>

Enables the search for frame indentifier.

Parameters:
<Identifier> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:SIDData <SearchName>,<IdentifierData>
SEARch:TRIGger:CAN:SIDData? <SearchName>

Enables the search for identifier and data.

Parameters:
<IdentifierData> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:SERRor <SearchName>,<ErrorCondition>
SEARch:TRIGger:CAN:SERRor? <SearchName>

Enables the search for a specified error.

Parameters:
<ErrorCondition> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:FDATa:STANdard <SearchName>,<Standard>
SEARch:TRIGger:CAN:FDATa:STANdard? <SearchName>

Selects the CAN standard: CAN, CAN FD, or Any.

Use "Any" to search on either CAN or CAN-FD frame. In this case, the search configuration provides all possible settings, for CAN as well as for CAN FD.

The setting is available in CAN FD option.

Parameters:
<Standard> ANY | CAN | CANFd
*RST: CAN

Parameters for setting and query:
<SearchName> String with the search name

SEARch:TRIGger:CAN:FTYPe <SearchName>,<FrameType>
SEARch:TRIGger:CAN:FTYPe? <SearchName>

Selects the CAN frame type to be searched for.

For data and remote frames, the identifier format has to be set with `SEARch:TRIGger:CAN:ITYPe` on page 1819.

Parameters:
<FrameType> ANY | DATA | REMote | ERRor | OVERload
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:ITYPe <SearchName>,<IdentifierType>
SEARch:TRIGger:CAN:ITYPe? <SearchName>

Selects the format of data and remote frames: 11 bit for CAN base frames, or 29 bits for CAN extended frames.

Parameters:
<IdentifierType> ANY | B11 | B29
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:ICONdition <SearchName>,<IdOperator>
SEARch:TRIGger:CAN:ICONdition? <SearchName>

Sets the operator to set a specific identifier or an identifier range.

Parameters:
<IdOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These condition require one identifier pattern to be set with `SEARch:TRIGger:CAN:IMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `SEARch:TRIGger:CAN:IMIN` and `SEARch:TRIGger:CAN:IMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:IMAX <SearchName>,<IdPatternTo>
SEARch:TRIGger:CAN:IMAX? <SearchName>

Sets the the end value of an identifier range if `SEARch:TRIGger:CAN:ICONdition` is set to `INRange` or `OORange`.

Parameters:
<IdPatternTo> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:IMIN <SearchName>,<IdPattern>
SEARch:TRIGger:CAN:IMIN? <SearchName>

Specifies a message identifier pattern, or sets the the start value of an identifier range.

Parameters:
<IdPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:CAN:DCONdition? <SearchName>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:
<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with `SEARch:TRIGger:CAN:DMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `SEARch:TRIGger:CAN:DMIN` and `SEARch:TRIGger:CAN:DMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:DMIN <SearchName>,<DataPattern>
SEARch:TRIGger:CAN:DMIN? <SearchName>

Specifies a data pattern, or sets the the start value of a data pattern range.

Parameters:
<DataPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:DMAX <SearchName>,<DataPatternTo>
SEARch:TRIGger:CAN:DMAX? <SearchName>

Sets the the end value of an data range if `SEARch:TRIGger:CAN:DCONdition` is set to `INRange` or `OORange`.

Parameters:
<DataPatternTo> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:DLCCondition <SearchName>,<DLCOperator>
SEARch:TRIGger:CAN:DLCCondition? <SearchName>

Operator to set the data length code for search.

Parameters:
<DLCOperator> EQUal | GETHan

*RST: GETHan

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:DLC <SearchName>,<WordCount>
SEARch:TRIGger:CAN:DLC? <SearchName>

Sets the Data Length Code, the number of data bytes to be found. For complete definition, set also the operator with `SEARch:TRIGger:CAN:DLCCondition`.

Parameters:

<WordCount>	Range:	1 to 8
	Increment:	1
	*RST:	1

Parameters for setting and query:
<SearchName>

SEARch:RESult:CAN:FRAMe<m>:NDBYtes? <SearchName>

Returns the number of data bytes defined by DLC. DLC and NDB are different in CAN FD for DLCs > 8.

See also: "Data setup: DLC, NDB, Transfer, Condition, Data min, Data max" on page 591.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<NDBytes>	Range:	1 to 64
	Increment:	1
	*RST:	1

Usage: Query only

SEARch:TRIGger:CAN:FDATa:DPOPerator <SearchName>,<DataPosOperator>
SEARch:TRIGger:CAN:FDATa:DPOPerator? <SearchName>

Sets the operator for the data position if DLC ≥ 9. You can define an exact position, or a position range.

The setting is available in CAN FD option.

Parameters:
<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe

ANY = OFF
The data position is not relevant for the search.

EQUal | GETHan
Equal, Greater or equal than. These conditions require one data position to be set with `SEARch:TRIGger:CAN:FDATa:DPOSition`.

INRange = RANGe
In range: Set the minimum and maximum value of the range with `SEARch:TRIGger:CAN:FDATa:DPOSition` and `SEARch:TRIGger:CAN:FDATa:DPTO`.

*RST: ANY

Parameters for setting and query:
<SearchName> String with the search name

SEARch:TRIGger:CAN:FDATa:DPOSition <SearchName>,<DataPosition>
SEARch:TRIGger:CAN:FDATa:DPOSition? <SearchName>

Defines the first possible start position of the data pattern.

The setting is available in CAN FD option.

Parameters:
<DataPosition> Range: 1 to 57
Increment: 1
*RST: 1

Parameters for setting and query:
<SearchName> String with the search name

SEARch:TRIGger:CAN:FDATa:DPTO <SearchName>,<DataPositionTo>
SEARch:TRIGger:CAN:FDATa:DPTO? <SearchName>

Defines the last possible start position of the data pattern if the position operator `SEARch:TRIGger:CAN:FDATa:DPOPerator` defines a range.

The setting is available in CAN FD option.

Parameters:
<DataPositionTo> Range: 8 to 64
Increment: 1
*RST: 8

Parameters for setting and query:
<SearchName> String with the search name

SEARch:TRIGger:CAN:ACKerror <SearchName>,<AckError>
SEARch:TRIGger:CAN:ACKerror? <SearchName>

Searches for acknowledgement errors. An acknowledgement error occurs when the transmitter does not receive an acknowledgment - a dominant bit during the Ack Slot.

Parameters:
<AckError> ON | OFF

*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:BITSterror <SearchName>,<BitStuffError>
SEARch:TRIGger:CAN:BITSterror? <SearchName>

Searches for bit stuffing errors.

Parameters:
<BitStuffError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:CRCerror <SearchName>,<ChecksumError>
SEARch:TRIGger:CAN:CRCerror? <SearchName>

Searches for errors in the Cyclic Redundancy Check.

Parameters:
<ChecksumError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:FORMerror <SearchName>,<FormError>
SEARch:TRIGger:CAN:FORMerror? <SearchName>

Searches for form errors. A form error occurs when a fixed-form bit field contains one or more illegal bits.

Parameters:
<FormError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:FDATa:SCERror <SearchName>,<StuffCountError>
SEARch:TRIGger:CAN:FDATa:SCERror? <SearchName>

Triggers on stuff count errors. A stuff count error occurs if th received stuff count value does not match the value calculated from the own stuff bit count.

Only relevant for CAN FD signals in ISO standard.

Parameters:
<StuffCountError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CAN:FDATa[:FDF] <SearchName>,<FDFBit>
SEARch:TRIGger:CAN:FDATa[:FDF]? <SearchName>

Sets the EDL bit (extended data length), which determines whether a frame is CAN or CAN-FD.

The setting is available in CAN FD option.

Parameters:
<FDFBit> ONE | ZERO | DC

ONE: CAN FD.
ZERO: CAN.
DC: don't care, the format is not relevant.

*RST: DC

Parameters for setting and query:
<SearchName> String with the search name

SEARch:TRIGger:CAN:FDATa:BRS <SearchName>,<BRSBit>
SEARch:TRIGger:CAN:FDATa:BRS? <SearchName>

Sets the bit rate switching bit for identifier and identifier + data searches.

The setting is available in CAN FD option.

Parameters:
<BRSBit> ONE | ZERO | DC

DC: Don't care

*RST: ONE

Parameters for setting and query:
<SearchName> String with the search name

SEARch:TRIGger:CAN:FDATa:ESI <SearchName>,<ESIBit>
SEARch:TRIGger:CAN:FDATa:ESI? <SearchName>

Sets the error state indicator bit for identifier and identifier + data searches.

The setting is available in CAN FD option.

Parameters:
<ESIBit> ONE | ZERO | DC

*RST: DC

Parameters for setting and query:
<SearchName> String with the search name

24.17.8.5 Search results

The search on decoded CAN data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 24.17.8.3, "Decode results", on page 1808.

SEARch:RESult:CAN:FCOunt? 1826
SEARch:RESult:CAN:FRAMe<m>:ACKValue? 1826
SEARch:RESult:CAN:FRAMe<m>:BSEPosition? 1827
SEARch:RESult:CAN:FRAMe<m>:BYTE<n>:STATe? 1827
SEARch:RESult:CAN:FRAMe<m>:BYTE<n>:VALue? 1827
SEARch:RESult:CAN:FRAMe<m>:ACKState? 1828
SEARch:RESult:CAN:FRAMe<m>:CSSTate? 1828
SEARch:RESult:CAN:FRAMe<m>:DLCState? 1828
SEARch:RESult:CAN:FRAMe<m>:IDSTate? 1828
SEARch:RESult:CAN:FRAMe<m>:CSValue? 1828
SEARch:RESult:CAN:FRAMe<m>:DATA? 1828
SEARch:RESult:CAN:FRAMe<m>:DLCValue? 1828
SEARch:RESult:CAN:FRAMe<m>:FERCause? 1829
SEARch:RESult:CAN:FRAMe<m>:IDTYpe? 1829
SEARch:RESult:CAN:FRAMe<m>:IDValue? 1829
SEARch:RESult:CAN:FDATa:FRAMe<m>:SCValue? 1830
SEARch:RESult:CAN:FDATa:FRAMe<m>:STANdard? 1830
SEARch:RESult:CAN:FRAMe<m>:STARt? 1830
SEARch:RESult:CAN:FRAMe<m>:STATus? 1830
SEARch:RESult:CAN:FRAMe<m>:STOP? 1831
SEARch:RESult:CAN:FRAMe<m>:SYMBol? 1831
SEARch:RESult:CAN:FRAMe<m>:TYPE? 1831

SEARch:RESult:CAN:FCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:

<Count>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:ACKValue? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<AckValue> Range: 0 to 1
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:BSEPosition? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<BitStuffErrorPos> Range: 0 to 5000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:BYTE<n>:STATe? <SearchName>

Suffix:

<m> *

<n> *

Query parameters:

<SearchName>

Return values:

<State> OK | ERRor | UNDF
*RST: OK

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:BYTE<n>:VALue? <SearchName>

Suffix:

<m> *

<n> *

Query parameters:

<SearchName>

Return values:

<Value> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:ACKState? <SearchName>
SEARch:RESult:CAN:FRAMe<m>:CSSTate? <SearchName>
SEARch:RESult:CAN:FRAMe<m>:DLCState? <SearchName>
SEARch:RESult:CAN:FRAMe<m>:IDSTate? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<IdentifierState> OK | ERRor | UNDF
*RST: OK

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:CSValue? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<ChecksumValue> Range: 0 to 2097151
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:DATA? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:DLCValue? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameDLCValue> Range: 0 to 15
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:FERCause? <SearchName>

Returns information on a form error, if the frame status query returned a FORM error.

See also: `SEARch:RESult:CAN:FRAMe<m>:STATus?` on page 1830.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FormErrorCause> NONE | CRCDerror | ACKDerror | FSBE | RESerror
See `BUS<m>:CAN:FRAMe<n>:FERCause?` on page 1815.
*RST: NONE

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:IDTYpe? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<IdentifierType> ANY | B11 | B29
*RST: B11

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:IDValue? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<IdentifierValue> Range: 0 to 536870911
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CAN:FDATa:FRAMe<m>:SCValue? <SearchName>

Returns the stuff bit count modulo 8.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<StuffCount> Range: 0 to 7
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CAN:FDATa:FRAMe<m>:STANdard? <SearchName>

Returns the CAN protocol standard: CAN or CAN FD.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Standard> CAN | CANFd
*RST: CAN

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:STATus? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameState> OK | FORM | BTST | CRC | CRCD | NOACk | ACKD | EOFD | CAERror | FCERor | INSufficient | SERRror | SFERror | SCERror | SAERror | SCAE | SCFE

See `BUS<m>:CAN:FRAMe<n>:STATus?` on page 1809.

*RST: OK

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:STOP? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:SYMBol? <SearchName>

Returns the symbolic label of the specified frame if the label list is enabled.

Suffix:
<m> *
Selects the number of the frame in the current acquisition, 1...n.

Query parameters:
<SearchName> String parameter that contains the search definition name

Return values:
<Label> Symbolic label (string)

Usage: Query only

SEARch:RESult:CAN:FRAMe<m>:TYPE? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameType> DATA | REMote | ERR | OVLD
*RST: DATA

Usage: Query only

24.17.8.6 Symbolic trigger, decode and search

- Symbolic trigger 1832
- Symbolic decode results 1833
- Symbolic search 1834

Symbolic trigger

TRIGger<m>:CAN:SYMBolic:MSGValue 1832
TRIGger<m>:CAN:SYMBolic:TSIGnals 1832
TRIGger<m>:CAN:SYMBolic:SIGValue 1832
TRIGger<m>:CAN:SYMBolic:DMAX 1833
TRIGger<m>:CAN:SYMBolic:DMIN 1833
TRIGger<m>:CAN:SYMBolic:SGEValue 1833

TRIGger<m>:CAN:SYMBolic:MSGValue <MessageName>

Sets the message to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same message if symbolic trigger and symbolic search is used at the same time.

See also: `SEARch:TRIGger:CAN:SYMBolic:MSGValue` on page 1835

Parameters:
<MessageName> String with the symbolic message name

TRIGger<m>:CAN:SYMBolic:TSIGnals <TriggerOnSignal>

Enables the trigger on a specific signal value that is part of the selected message.

Parameters:
<TriggerOnSignal> ON | OFF
*RST: OFF

TRIGger<m>:CAN:SYMBolic:SIGValue <SignalName>

Sets the signal name to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same signal if symbolic trigger and symbolic search is used at the same time.

See also: `SEARch:TRIGger:CAN:SYMBolic:SIGValue` on page 1835

Parameters:
<SignalName> String with the signal name as defined in the DBC file.

TRIGger<m>:CAN:SYMBolic:DMAX <DataPatternTo>

Defines the maximum data value of the signal.

This value is required to specify a range if condition `INRange | OORange` is set with `TRIGger<m>:CAN:DCONdition`.

Parameters:
<DataPatternTo> Decimal representation of the data pattern

Range: -100E+24 to 100E+24
Increment: 1
*RST: 1

TRIGger<m>:CAN:SYMBolic:DMIN <DataPattern>

Defines the minimum data value of the signal.

To set the condition, use `TRIGger<m>:CAN:DCONdition`.

Parameters:
<DataPattern> Decimal representation of the data pattern

Range: -100E+24 to 100E+24
Increment: 1
*RST: 0

TRIGger<m>:CAN:SYMBolic:SGEValue <SignalEnumValue>

Sets a symbolic data value for signals with enumerated values.

Parameters:
<SignalEnumValue> Numeric value according to the value definition in the DBC file

Example: Definition line in DBC file:

```
VAL_ 2175091489 Gear 0 "Idle" 1 "Gear_1" 2 "Gear_2" 3 "Gear_3"
4 "Gear_4" 5 "Gear_5" ;
```

Search for "Gear_4"

```
TRIGger:CAN:SYMBolic:SGEValue 4
```

Symbolic decode results

BUS<m>:CAN:FRAMe<n>:SDATa?

Returns the complete symbolic data of the selected frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<SymbolicData> String with comma-separated list of symbolic data

Example:
```
BUS:CAN:FRAMe9:SDATa?
<-- [sym] 325 kW, 0x0A, 423 N, 174 l, Running, 90 degC, 0x06, 437 rpm
```
Returns the symbolic results of the 9th frame.

Usage: Query only

Symbolic search

SEARch:RESult:CAN:FRAMe<m>:SDATa? 1834
SEARch:TRIGger:CAN:SSYMbolic 1834
SEARch:TRIGger:CAN:SYMBolic:MSGValue 1835
SEARch:TRIGger:CAN:SYMBolic:SSIGnals 1835
SEARch:TRIGger:CAN:SYMBolic:SIGValue 1835
SEARch:TRIGger:CAN:SYMBolic:DMIN 1836
SEARch:TRIGger:CAN:SYMBolic:DMAX 1836
SEARch:TRIGger:CAN:SYMBolic:SGEValue 1836

SEARch:RESult:CAN:FRAMe<m>:SDATa? <SearchName>

Returns the symbolic data of the selected result frame.

Suffix:
<m> *
Sets the index of the search result frame.

Query parameters:
<SearchName> String that contains the search definition name

Return values:
<SymbolicData> String with comma-separated list of symbolic data

Example:
```
SEARch:RESult:CAN:FRAMe:SDATa? 'Search1'
<-- [sym] 325 kW, 0x0A, 423 N, 174 l, Running, 90 degC, 0x06, 437 rpm
```
Returns the symbolic results of the first search result.

Usage: Query only

SEARch:TRIGger:CAN:SSYMbolic <SearchName>,<CheckSymbolic>
SEARch:TRIGger:CAN:SSYMbolic? <SearchName>

Enables the symbolic search and disables all other search criteria.

Parameters:
<CheckSymbolic> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName> String that contains the search definition name

SEARch:TRIGger:CAN:SYMBolic:MSGValue <SearchName>, <MessageName>
SEARch:TRIGger:CAN:SYMBolic:MSGValue? <SearchName>

Sets the message to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same message if symbolic trigger and symbolic search is used at the same time.

See also: `TRIGger<m>:CAN:SYMBolic:MSGValue` on page 1832

Parameters:
<MessageName> String that contains the symbolic message name

Parameters for setting and query:
<SearchName> String that contains the search definition name

Example: `SEARch:TRIGger:CAN:SYMBolic:MSGValue "Search1","EngineData"`

SEARch:TRIGger:CAN:SYMBolic:SSIGnals <SearchName>,<SymbolicSearch>
SEARch:TRIGger:CAN:SYMBolic:SSIGnals? <SearchName>

Enables the search for symbolic values if DBC label list file is loaded and applied. Symbolic search disables all other search criteria.

Parameters:
<SymbolicSearch> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName> String that contains the search definition name

SEARch:TRIGger:CAN:SYMBolic:SIGValue <SearchName>, <SignalName>
SEARch:TRIGger:CAN:SYMBolic:SIGValue? <SearchName>

Sets the signal name to be triggered or searched for.

The setting is used by symbolic trigger and symbolic search. Thus, you always trigger on and search for the same signal if symbolic trigger and symbolic search is used at the same time.

See also: `TRIGger<m>:CAN:SYMBolic:SIGValue` on page 1832

Parameters:
<SignalName> String that contains the symbolic signal name

Parameters for setting and query:
<SearchName> String that contains the search definition name

Example: `SEARch:TRIGger:CAN:SYMBolic:SIGValue "Search1","EngForce"`

SEARch:TRIGger:CAN:SYMBolic:DMIN <SearchName>,<DataPattern>
SEARch:TRIGger:CAN:SYMBolic:DMIN? <SearchName>

Defines the minimum data pattern in a symbolic search.

To set the condition, use `SEARch:TRIGger:CAN:DCONdition`.

Parameters:

<DataPattern>	Range:	-100E+24 to 100E+24
	Increment:	1
	*RST:	0

Parameters for setting and query:

<SearchName> String that contains the search definition name

SEARch:TRIGger:CAN:SYMBolic:DMAX <SearchName>,<DataPatternTo>
SEARch:TRIGger:CAN:SYMBolic:DMAX? <SearchName>

Defines the maximum data pattern of the signal in a symbolic search.

This value is required to specify a range if condition `INRange | OORange` is set with `SEARch:TRIGger:CAN:DCONdition` on page 1820.

Parameters:

<DataPatternTo>	Range:	-100E+24 to 100E+24
	Increment:	1
	*RST:	1

Parameters for setting and query:

<SearchName> String that contains the search definition name

SEARch:TRIGger:CAN:SYMBolic:SGEValue <SearchName>, <SignalEnumValue>
SEARch:TRIGger:CAN:SYMBolic:SGEValue? <SearchName>

Sets a symbolic data value for signals with enumerated values.

Parameters:

<SignalEnumValue> Numeric value according to the value definition in the DBC file

Parameters for setting and query:

<SearchName> String that contains the search definition name

Example: Definition line in DBC file:

```
VAL_ 2175091489 Gear 0 "Idle" 1 "Gear_1" 2 "Gear_2"
3 "Gear_3" 4 "Gear_4" 5 "Gear_5" ;
```

Search for "Gear_3"

```
SEARch:TRIGger:CAN:SYMBolic:SGEValue "Search1",3
```

24.17.9 LIN (option R&S RTO6-K520)

- Configuration....1837
- Trigger....1839
- Decode results....1845
- LIN search settings....1852
- LIN search results....1859

24.17.9.1 Configuration

BUS<m>:LIN:DATA:SOURce....1837
BUS<m>:LIN:DATA:THReshold....1837
BUS<m>:LIN:TECHnology....1838
BUS<m>:LIN:BITRate....1838
BUS<m>:LIN:POLarity....1838
BUS<m>:LIN:STANdard....1839

BUS<m>:LIN:DATA:SOURce <DataSource>

Sets the waveform of the data line.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<DataSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

See Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C1W1

Usage: Asynchronous command

BUS<m>:LIN:DATA:THReshold <Threshold>

Sets a user-defined threshold value.

Alternatively, you can set the threshold according to the signal technology with `BUS<m>:LIN:TECHnology`.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Threshold> Range: -12 to 12
Increment: 0.1
*RST: 0
Default unit: V

BUS<m>:LIN:TECHnology <Technology>

Sets the threshold voltage as defined for various signal technologies.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Technology> V15 | V25 | V35 | V6 | V9 | MANual

V15
1.5 Volt (TTL)

V25
2.5 Volt (CMOS 5.0 V)

V35 | V6 | V9
3.5 V (7 V supply), 6.0 V (12 V supply), 9.0 V (18 V supply) respectively

MANual
Manual setting of user-defined values with `BUS<m>:LIN:DATA:THReshold`.

*RST: V35

BUS<m>:LIN:BITRate <Bitrate>

Sets the number of transmitted bits per second.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Bitrate> Range: 1000 to 20000
Increment: 1
*RST: 9600
Default unit: bps

BUS<m>:LIN:POLarity <Polarity>

Defines the idle state of the bus. The idle state is the rezessive state and corresponds to a logic 1.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Polarity> IDLLow | IDLHigh

*RST: IDLHigh

BUS<m>:LIN:STANdard <Standard>

Selects the version of the LIN standard.

See also: "LIN standard" on page 606

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Standard> V1X | V2X | J2602 | AUTO
*RST: AUTO

24.17.9.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:LIN:TYPE 1839
TRIGger<m>:LIN:ICONdition 1840
TRIGger<m>:LIN:IMIN 1841
TRIGger<m>:LIN:IMAX 1841
TRIGger<m>:LIN:DCONdition 1841
TRIGger<m>:LIN:DMIN 1841
TRIGger<m>:LIN:DMAX 1841
TRIGger<m>:LIN:BORDer 1842
TRIGger<m>:LIN:DLECondition 1842
TRIGger<m>:LIN:DLENgth 1842
TRIGger<m>:LIN:IDOR<n>:ENABle 1843
TRIGger<m>:LIN:IDOR<n>[:VALue] 1843
TRIGger<m>:LIN:SYERror 1843
TRIGger<m>:LIN:IPERror 1843
TRIGger<m>:LIN:CHKSerror 1844
TRIGger<m>:LIN:ERRPattern 1844
TRIGger<m>:LIN:CRCDatalen 1844
TRIGger<m>:LIN:STANdard 1845

TRIGger<m>:LIN:TYPE <Type>

Selects the trigger type for LIN analysis.

See: "Trigger type" on page 611.

Parameters:
<Type> SYNC | ID | IDOR | IDDT | WKFR | ERRC

SYNC
Start of the frame, triggers on the stop bit of the sync field.

ID
Sets the trigger to one specific identifier or an identifier range.
To set the identifier, use `TRIGger<m>:LIN:ICONdition`, `TRIGger<m>:LIN:IMIN` on page 1841, and `TRIGger<m>:LIN:IMAX` on page 1841.

IDOR
Triggers on an OR combination with up to four identifier conditions. For each identifier condition, enable it with `TRIGger<m>:LIN:IDOR<n>:ENABle` and set the value with `TRIGger<m>:LIN:IDOR<n>[:VALue]`

IDDT
Combination of identifier and data conditions
To set the identifier condition, use `TRIGger<m>:LIN:ICONdition`, `TRIGger<m>:LIN:IMIN`, and `TRIGger<m>:LIN:IMAX`.
To set the data condition, use `TRIGger<m>:LIN:DCONdition`, `TRIGger<m>:LIN:DMIN`, and `TRIGger<m>:LIN:DMAX`.

WKFR
Wakeup frame

ERRC
Error condition. Define the error types with `TRIGger<m>:LIN:CHKSerror` on page 1844, `TRIGger<m>:LIN:IPERror`, and `TRIGger<m>:LIN:SYERror`

*RST: SYNC

TRIGger<m>:LIN:ICONdition <IdOperator>

Sets the operator to set a specific identifier or an identifier range.

Parameters:

<IdOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These condition require one identifier pattern to be set with `TRIGger<m>:LIN:IMIN`

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:LIN:IMIN` and `TRIGger<m>:LIN:IMAX`

*RST: EQUal

TRIGger<m>:LIN:IMIN <IdPattern>

Specifies a slave identifier pattern, or sets the the start value of an identifier range.

Parameters:

<IdPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

TRIGger<m>:LIN:IMAX <IdPatternTo>

Sets the the end value of an identifier range if `TRIGger<m>:LIN:ICONdition` is set to `INRange` or `OORange`.

Parameters:

<IdPatternTo> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

TRIGger<m>:LIN:DCONdition <DataOperator>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with `TRIGger<m>:LIN:DMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:LIN:DMIN` and `TRIGger<m>:LIN:DMAX`

*RST: EQUal

TRIGger<m>:LIN:DMIN <DataPattern>

Specifies a data pattern, or sets the the start value of a data pattern range.

Parameters:

<DataPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:LIN:DMAX <DataPatternTo>

Sets the the end value of an data range if `TRIGger<m>:LIN:DCONdition` is set to `INRange` or `OORange`.

Parameters:

<DataPatternTo> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:LIN:BORDer <Endianness>

Sets the byte order (endianness) of the data transfer.

According to the standard, LIN data is transmitted in little endian transfer order.

Parameters:

<Endianness> BENDian | LENDian

BENDian
Big endian, data is analyzed and evaluated in the order of reception.

LENDian
Little endian, the instrument reads the complete data, reverses the byte order of the data, and compares it with the specified data word.

*RST: BENDian

TRIGger<m>:LIN:DLECondition <DLCOperator>

Operator to set the data length for triggering on LIN data.

For Big Endian transfer direction, you can trigger on a number of bytes less than the data length of the frame, that means, on the first bytes that are transmitted. For Little Endian transfer direction, the exact number of data bytes in the frame must be set.

Example: The data word to be sent is *12 34 56*, and it is sent little endian by the LIN node. With Data length ≥ 2 and Transfer = Big endian, you trigger on the data of the first two bytes, that is *56 34*. With Data length = 3 and Transfer = Little endian, you trigger on the required data word *12 34 56*.

The number of data bytes to be found is set with `TRIGger<m>:LIN:DLENgth` on page 1842.

See also:`TRIGger<m>:LIN:BORDer` on page 1842 .

Parameters:

<DLCOperator> EQUal | GETHan

For little endian transfer direction, EQUal must be set.

*RST: GETHan

TRIGger<m>:LIN:DLENgth <WordCount>

Sets the length of the bit pattern to be found, in bytes. For "Big Endian" transfer direction, you can trigger on a number of bytes less than the data length of the frame, that means, on the beginning of the data pattern. For "Little Endian" transfer direction, the exact number of data bytes in the frame must be set.

For complete definition, set also the operator with `TRIGger<m>:LIN:DLECondition` on page 1842.

Parameters:

<WordCount> Range: 1 to 8
Increment: 1
*RST: 1

TRIGger<m>:LIN:IDOR<n>:ENABle <UseIdentifier>

Includes the indicated IDOR address in the "identifier OR" trigger condition.

Suffix:

<n> 1..4
Index of the identifier in an "identifier OR" condition

Parameters:

<UseIdentifier> ON | OFF
*RST: OFF

TRIGger<m>:LIN:IDOR<n>[:VALue] <IdPattern>

Defines the pattern of the indicated IDOR identifier in the "identifier OR" trigger condition.

Suffix:

<n> 1..4
Index of the identifier in an "identifier OR" condition

Parameters:

<IdPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The parameter accepts the bit value X (don't care).

TRIGger<m>:LIN:SYERror <SyncError>

Triggers if a synchronization error occurs.

The trigger type has to be set before: `TRIGger<m>:LIN:TYPE` to `ERRC`.

Parameters:

<SyncError> ON | OFF
*RST: ON

TRIGger<m>:LIN:IPERror <IdParityError>

Triggers if an error occurs in the identifier parity bits. These are the bits 6 and 7 of the identifier.

The trigger type has to be set before: `TRIGger<m>:LIN:TYPE` to `ERRC`.

Parameters:
<IdParityError> ON | OFF
*RST: ON

TRIGger<m>:LIN:CHKSerror <ChecksumError>

Triggers on checksum errors according to the LIN standard set with `BUS<m>:LIN:STANdard`.

The trigger type has to be set before: `TRIGger<m>:LIN:TYPE` to `ERRC`.

The frame identifier must be set with `TRIGger<m>:LIN:ERRPattern` on page 1844 and the data length with `TRIGger<m>:LIN:CRCDatalen` on page 1844.

Parameters:
<ChecksumError> ON | OFF
*RST: ON

TRIGger<m>:LIN:ERRPattern <ErrorPattern>

Sets the frame identifier to trigger on a checksum error with `TRIGger<m>:LIN:CHKSerror` on page 1844.

Parameters:
<ErrorPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.
Possible values depend on `TRIGger<m>:LIN:CRCDatalen`.
Defining don't care bits 'X' in the ERRP bit string resets CRCDatalen to 0. When CRCDatalen is different than 0, then all the bits in ERRP must be 1 or 0, and X bits are set to 0.

Example:

```
FORM:BPAT STRG
TRIG1:LIN:ERRP '1X0'
TRIG1:LIN:ERRP?
<-- 1X0XXX
TRIGger:LIN:CRCDatalen?
<-- 0
TRIGger:LIN:CRCDatalen 4
TRIG1:LIN:ERRP?
<-- 100000
TRIG1:LIN:ERRP '00x11'
TRIG1:LIN:ERRP?
<-- 00X11X
TRIGger:LIN:CRCDatalen?
<-- 0
```

TRIGger<m>:LIN:CRCDatalen <CRCDataLength>

Sets the number of data bytes to trigger on CRC errors (`TRIGger<m>:LIN:TYPE` is set to ERRC and `TRIGger<m>:LIN:CHKSerror` is set ON.)

Parameters:

<CRCDataLength> Values ≠0 restrict allowed bit values in `TRIGger<m>:LIN:ERRPattern` to 0 and 1.

Range: 0 to 8
Increment: 1
*RST: 0

TRIGger<m>:LIN:STANdard <LINStandard>

Sets the LIN standard to trigger on CRC errors (`TRIGger<m>:LIN:TYPE` is set to ERRC and `TRIGger<m>:LIN:CHKSerror` is set ON.)

See also: "LIN standard" on page 606.

Parameters:

<LINStandard> V1X | V2X | J2602 | AUTO

*RST: AUTO

24.17.9.3 Decode results

To load and acivate a label list, use:

- `BUS<m>:NEWList` on page 1672
- `BUS<m>:SYMBols` on page 1672

BUS<m>:LIN:FCOunt?.....1845
BUS<m>:LIN:FRAMe<n>:STATus?.....1846
BUS<m>:LIN:FRAMe<n>:STARt?.....1846
BUS<m>:LIN:FRAMe<n>:STOP?.....1846
BUS<m>:LIN:FRAMe<n>:SDATa?.....1847
BUS<m>:LIN:FRAMe<n>:SDEXport?.....1847
BUS<m>:LIN:FRAMe<n>:SYMBol?.....1847
BUS<m>:LIN:FRAMe<n>:VERSion?.....1848
BUS<m>:LIN:FRAMe<n>:DATA?.....1848
BUS<m>:LIN:FRAMe<n>:IDSTate?.....1848
BUS<m>:LIN:FRAMe<n>:IDValue?.....1849
BUS<m>:LIN:FRAMe<n>:IDPValue?.....1849
BUS<m>:LIN:FRAMe<n>:SYSTate?.....1850
BUS<m>:LIN:FRAMe<n>:CSSTate?.....1850
BUS<m>:LIN:FRAMe<n>:CSValue?.....1850
BUS<m>:LIN:FRAMe<n>:BITRate?.....1851
BUS<m>:LIN:FRAMe<n>:BYTE<o>:STATe?.....1851
BUS<m>:LIN:FRAMe<n>:BYTE<o>:VALue?.....1852

BUS<m>:LIN:FCOunt?

Returns the number of decoded frames.

Suffix:

<m> 1..4
Selects the serial bus.

Return values:

<Count> Total number of decoded frames.

Usage: Query only

BUS<m>:LIN:FRAMe<n>:STATus?

Returns the overall state of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameState> OK | UART | CHCKsum | VERS | LENer | SPERror | PRERror | SYERror | WAKeup | CPERror | INSufficient | INComplete

UART: at least one UART error occured. LIN uses UART words without parity bit.
CHCKsum: checksum error
VERS: the version of the LIN standard is not valid
LENer: unexpected length
SPERror: stop error
PRERror: parity error in identifier
SYERror: synchronization error
WAKeup: the frame is a wakeup frame
CPERror: parity error and checksum error
INSufficient: the frame is not completely contained in the acquisition. The decoded part of the frame is valid.
INComplete: the frame is missing some parts

*RST: OK

Usage: Query only

BUS<m>:LIN:FRAMe<n>:STARt?
BUS<m>:LIN:FRAMe<n>:STOP?

Returns the start time and stop time of the selected frame, respectively.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Start>, <Stop> Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:LIN:FRAMe<n>:SDATa?

Returns the complete symbolic data of the frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<SymbolicData>

Usage: Query only

BUS<m>:LIN:FRAMe<n>:SDEXport?

Returns the symbolic data of the frame in export format.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<SymbolicData>

Usage: Query only

BUS<m>:LIN:FRAMe<n>:SYMBol?

Returns the symbolic label of the specified frame if the label list is enabled.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the number of the frame in the current acquisition, 1...n.

Return values:

<Label> String with symbolic name of the identifier

Example: `BUS:LIN:FRAMe2:SYMBol?`
Response: `Temperature`

Usage: Query only

BUS<m>:LIN:FRAMe<n>:VERSion?

Returns the version of the LIN standard for the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameVersion> V1X | V2X | UNK

UNK: Unknown

*RST: UNK

Usage: Query only

BUS<m>:LIN:FRAMe<n>:DATA?

Returns the data bytes of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Data> Comma-separated list of integer values (N, D1, D2,…, DN). N is the number of bytes in the frame, and D1…DN are the values of the bytes.

Example:

```
BUS:LIN:FRAMe4:DATA?
<-- 4,118,39,71,123
```

Usage: Query only

BUS<m>:LIN:FRAMe<n>:IDSTate?

Returns the identifier state of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<IdentifierState> OK | STERror | SPERror | PRERror | UVAL | NOEXists | INSufficient

STERror: start error

SPERror: stop error
PRERror: parity error
UVAL: unexpected value
NOEXists: byte does not exist
INSufficient: the frame is not completely contained in the acquisition. The decoded part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:LIN:FRAMe<n>:IDValue?

Returns the identifier value of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<IdentifierValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: 0 to 63
*RST: 0

Usage: Query only

BUS<m>:LIN:FRAMe<n>:IDPValue?

Returns the value of the identifier parity bits of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<IdtfParityValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: 0 to 3
*RST: 0

Usage: Query only

BUS<m>:LIN:FRAMe<n>:SYSTate?

Returns the state of the sync field for the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<SyncState> OK | STERror | SPERror | UVAL | NOEXists | INSufficient

STERror: start error
SPERror: stop error
UVAL: unexpected value
NOEXists: byte does not exist
INSufficient: the frame is not completely contained in the acquisition. The decoded part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:LIN:FRAMe<n>:CSSTate?

Returns the checksum state of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<ChecksumState> OK | STERror | SPERror | UVAL | NOEXists | INSufficient

STERror: start error
SPERror: stop error
UVAL: unexpected value
NOEXists: byte does not exist
INSufficient: the frame is not completely contained in the acquisition. The decoded part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:LIN:FRAMe<n>:CSValue?

Returns the checksum value of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<ChecksumValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: 0 to 255
*RST: 0

Usage: Query only

BUS<m>:LIN:FRAMe<n>:BITRate?

Returns the primary bit rate.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<PrimaryBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

BUS<m>:LIN:FRAMe<n>:BYTE<o>:STATe?

Returns the state of the specified byte.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte number.

Return values:

<ByteState> OK | STERror | SPERror | UVAL | NOEXists | INSufficient

STERror: start error
SPERror: stop error
UVAL: unexpected value

NOEXists: byte does not exist
INSufficient: the frame is not completely contained in the acquisition. The decoded part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:LIN:FRAMe<n>:BYTE<o>:VALue?

Returns the value of the specified byte.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the byte.

Return values:

<ByteValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: 0 to 255
*RST: 0

Usage: Query only

24.17.9.4 LIN search settings

In search setup commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name. The commands are similar to LIN trigger commands.

SEARch:TRIGger:LIN:SSOFrame....1853
SEARch:TRIGger:LIN:SFIDentifier....1853
SEARch:TRIGger:LIN:IDENtifieror....1853
SEARch:TRIGger:LIN:SIDData....1853
SEARch:TRIGger:LIN:SERRor....1854
SEARch:TRIGger:LIN:WUFRame....1854
SEARch:TRIGger:LIN:ICONdition....1854
SEARch:TRIGger:LIN:IMIN....1855
SEARch:TRIGger:LIN:IMAX....1855
SEARch:TRIGger:LIN:IDOR<m>:ENABle....1855
SEARch:TRIGger:LIN:IDOR<m>[:VALue]....1855
SEARch:TRIGger:LIN:DCONdition....1856
SEARch:TRIGger:LIN:DMIN....1856
SEARch:TRIGger:LIN:DMAX....1856
SEARch:TRIGger:LIN:BORDer....1857
SEARch:TRIGger:LIN:DLECondition....1857

SEARch:TRIGger:LIN:DLENgth....1858
SEARch:TRIGger:LIN:IPERror....1858
SEARch:TRIGger:LIN:SYERror....1858
SEARch:TRIGger:LIN:CHKSerror....1858
SEARch:TRIGger:LIN:ERRPattern....1859
SEARch:TRIGger:LIN:CRCDatalen....1859
SEARch:TRIGger:LIN:STANdard....1859

SEARch:TRIGger:LIN:SSOFrame <SearchName>,<FrameStart>
SEARch:TRIGger:LIN:SSOFrame? <SearchName>

Enables the search for the stop bit of the sync field, which marks the frame start.

Parameters:
<FrameStart> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:SFIDentifier <SearchName>,<Identifier>
SEARch:TRIGger:LIN:SFIDentifier? <SearchName>

Enables the search for one specific identifier or an identifier range.

Parameters:
<Identifier> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:IDENtifieror <SearchName>,<IdentifierOr>
SEARch:TRIGger:LIN:IDENtifieror? <SearchName>

Enables the search for one to four address conditions.

Parameters:
<IdentifierOr> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:SIDData <SearchName>,<IdentifierData>
SEARch:TRIGger:LIN:SIDData? <SearchName>

Enables the search for a combination of identifier and data conditions.

Parameters:
<IdentifierData> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:SERRor <SearchName>,<ErrorCondition>
SEARch:TRIGger:LIN:SERRor? <SearchName>

Enables the search for various errors in the frame.

Parameters:
<ErrorCondition> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:WUFRame <SearchName>,<WakeUpFrame>
SEARch:TRIGger:LIN:WUFRame? <SearchName>

Enables the search for wakeup frames.

Parameters:
<WakeUpFrame> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:ICONdition <SearchName>,<IdOperator>
SEARch:TRIGger:LIN:ICONdition? <SearchName>

Sets the operator to define a specific identifier or an identifier range.

Parameters:
<IdOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These operators require one identifier pattern to be set with `SEARch:TRIGger:LIN:IMIN`

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `SEARch:TRIGger:LIN:IMIN` and `SEARch:TRIGger:LIN:IMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:IMIN <SearchName>,<IdPattern>
SEARch:TRIGger:LIN:IMIN? <SearchName>

Specifies a slave identifier pattern, or sets the the start value of an identifier range.

Parameters:
<IdPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:IMAX <SearchName>,<IdPatternTo>
SEARch:TRIGger:LIN:IMAX? <SearchName>

Sets the the end value of an identifier range if `SEARch:TRIGger:LIN:ICONdition` is set to `INRange` or `OORange`.

Parameters:
<IdPatternTo> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:IDOR<m>:ENABle <SearchName>,<UseIdentifier>
SEARch:TRIGger:LIN:IDOR<m>:ENABle? <SearchName>

Includes the indicated IDOR address in the "identifier OR" search.

Suffix:
<m> 1..4
Index of the identifier in an "identifier OR" condition

Parameters:
<UseIdentifier> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:IDOR<m>[:VALue] <SearchName>,<IdPattern>
SEARch:TRIGger:LIN:IDOR<m>[:VALue]? <SearchName>

Defines the pattern of the indicated IDOR identifier in the "identifier OR" trigger condition.

Suffix:

<m> 1..4
Index of the identifier in an "identifier OR" condition

Parameters:

<IdPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:LIN:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:LIN:DCONdition? <SearchName>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with `SEARch:TRIGger:LIN:DMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `SEARch:TRIGger:LIN:DMIN` and `SEARch:TRIGger:LIN:DMAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:LIN:DMIN <SearchName>,<DataPattern>
SEARch:TRIGger:LIN:DMIN? <SearchName>

Specifies a data pattern, or sets the the start value of a data pattern range.

Parameters:

<DataPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:LIN:DMAX <SearchName>,<DataPatternTo>
SEARch:TRIGger:LIN:DMAX? <SearchName>

Sets the the end value of an identifier range if `SEARch:TRIGger:LIN:DCONdition` is set to `INRange` or `OORange`.

Parameters:
<DataPatternTo> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:BORDer <SearchName>,<Endianness>
SEARch:TRIGger:LIN:BORDer? <SearchName>

Sets the byte order (endianness) of the data transfer.

Parameters:
<Endianness> BENDian | LENDian

BENDian
Big endian, data is analyzed and evaluated in the order of reception.

LENDian
Little endian, the instrument reads the complete data, reverses the byte order of the data, and compares it with the specified data word.

*RST: BENDian

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:DLECondition <SearchName>,<DLCOperator>
SEARch:TRIGger:LIN:DLECondition? <SearchName>

Operator to set the data length for search on LIN data.

For Big Endian transfer direction, you can trigger on a number of bytes less than the data length of the frame, that means, on the first bytes that are transmitted. For Little Endian transfer direction, the exact number of data bytes in the frame must be set.

Example: The data word to be sent is *12 34 56*, and it is sent little endian by the LIN node. With Data length ≥ 2 and Transfer = Big endian, you trigger on the data of the first two bytes, that is *56 34*. With Data length = 3 and Transfer = Little endian, you trigger on the required data word *12 34 56*.

The number of data bytes to be found is set with `SEARch:TRIGger:LIN:DLENgth`.

See also: `SEARch:TRIGger:LIN:BORDer` on page 1857.

Parameters:
<DLCOperator> EQUal | GETHan

For little endian transfer direction, EQUal must be set.

*RST: GETHan

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:DLENgth <SearchName>,<WordCount>
SEARch:TRIGger:LIN:DLENgth? <SearchName>

Sets the length of the bit pattern to be found, in bytes.

For complete definition, set the operator using `SEARch:TRIGger:LIN:DLECondition`, and the transfer direction with `SEARch:TRIGger:LIN:BORDer`.

Parameters:

<WordCount>	Range:	1 to 8
	Increment:	1
	*RST:	1

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:IPERror <SearchName>,<IdParityError>
SEARch:TRIGger:LIN:IPERror? <SearchName>

Searches for errors in the identifier parity bits. These are the bits 6 and 7 of the identifier.

Parameters:

<IdParityError>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:SYERror <SearchName>,<SyncError>
SEARch:TRIGger:LIN:SYERror? <SearchName>

Searches for synchronization errors.

Parameters:

<SyncError>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:CHKSerror <SearchName>,<ChecksumError>
SEARch:TRIGger:LIN:CHKSerror? <SearchName>

Searches for checksum errors according to the LIN standard.

Use the following commands to configure the checksum error search:

- `SEARch:TRIGger:LIN:ERRPattern` on page 1859
- `SEARch:TRIGger:LIN:CRCDatalen` on page 1859
- `SEARch:TRIGger:LIN:STANdard` on page 1859

Parameters:
<ChecksumError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:ERRPattern <SearchName>,<ErrorPattern>
SEARch:TRIGger:LIN:ERRPattern? <SearchName>

Sets the frame identifier to search for a checksum error.

Parameters:
<ErrorPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:CRCDatalen <SearchName>,<CRCDataLength>
SEARch:TRIGger:LIN:CRCDatalen? <SearchName>

Sets the number of data bytes search for CRC errors.

Parameters:
<CRCDataLength> Range: 0 to 8
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:LIN:STANdard <SearchName>,<LINStandard>
SEARch:TRIGger:LIN:STANdard? <SearchName>

Sets the LIN standard to search for CRC errors.

Parameters:
<LINStandard> V1X | V2X | J2602 | AUTO
*RST: AUTO

Parameters for setting and query:
<SearchName>

24.17.9.5 LIN search results

The search on decoded LIN data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 24.17.9.3, "Decode results", on page 1845.

SEARch:RESult:LIN:FCOunt?.....1860
SEARch:RESult:LIN:FRAMe<m>:STATus?.....1860
SEARch:RESult:LIN:FRAMe<m>:STARt?.....1860
SEARch:RESult:LIN:FRAMe<m>:STOP?.....1861
SEARch:RESult:LIN:FRAMe<m>:DATA?.....1861
SEARch:RESult:LIN:FRAMe<m>:SDATa?.....1861
SEARch:RESult:LIN:FRAMe<m>:SDEXport?.....1862
SEARch:RESult:LIN:FRAMe<m>:CSSTate?.....1862
SEARch:RESult:LIN:FRAMe<m>:CSValue?.....1862
SEARch:RESult:LIN:FRAMe<m>:IDSTate?.....1863
SEARch:RESult:LIN:FRAMe<m>:IDValue?.....1863
SEARch:RESult:LIN:FRAMe<m>:IDPValue?.....1863
SEARch:RESult:LIN:FRAMe<m>:SYMBol?.....1864
SEARch:RESult:LIN:FRAMe<m>:SYSTate?.....1864
SEARch:RESult:LIN:FRAMe<m>:VERSion?.....1864
SEARch:RESult:LIN:FRAMe<m>:BYTE<n>:STATe?.....1864
SEARch:RESult:LIN:FRAMe<m>:BYTE<n>:VALue?.....1865

SEARch:RESult:LIN:FCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:

<Count>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:STATus? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<State>	OK | UART | CHCKsum | VERS | LENer | SPERror | PRERror | SYERror | WAKeup | CPERror | INSufficient | INComplete	
	*RST:	OK

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:STOP? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:DATA? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data> Data bytes in the Frame

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:SDATa? <SearchName>

Returns the symbolic data of the frame.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SymbolicData>

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:SDEXport? <SearchName>

Returns the symbolic data of the frame for the export format.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SymbolicData>

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:CSSTate? <SearchName>

Returns the status of the frame checksum.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<ChecksumState> OK | STERror | SPERror | PRERror | UVAL | NOEXists | INSufficient

STERror: StarT ERror, incorrect start bit
SPERror: StoP ERror, incorrect stop bit
PRERror:PaRity ERror, incorrect parity bit.
UVAL: unexpected value
NOEXists: byte does not exist

*RST: OK

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:CSValue? <SearchName>

Returns the checksum value.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<ChecksumValue> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:IDSTate? <SearchName>

Returns the status of the identifier.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<IdentifierState> OK | STERror | SPERror | PRERror | UVAL | NOEXists | INSufficient

STERror: StarT ERror, incorrect start bit
SPERror: StoP ERror, incorrect stop bit
PRERror:PaRity ERror, incorrect parity bit.
UVAL: unexpected value
NOEXists: byte does not exist

*RST: OK

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:IDValue? <SearchName>

Returns the value of the identifier.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<IdentifierValue> Range: 0 to 63
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:IDPValue? <SearchName>

Returns the value of the identifier parity bit.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<IdParityValue> Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:SYMBol? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Label>

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:SYSTate? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SyncState> OK | STERror | SPERror | PRERror | UVAL | NOEXists | INSufficient

*RST: OK

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:VERSion? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Version> V1X | V2X | UNK

*RST: UNK

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:BYTE<n>:STATe? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<ByteState> OK | STERror | SPERror | PRERror | UVAL | NOEXists | INSufficient
*RST: OK

Usage: Query only

SEARch:RESult:LIN:FRAMe<m>:BYTE<n>:VALue? <SearchName>

Suffix:
<m> *
<n> *

Query parameters:
<SearchName>

Return values:
<ByteValue> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

24.17.10 Flexray (option R&S RTO6-K520)

- Configuration....1865
- Trigger....1870
- Decode results....1878
- Search settings....1884
- Search results....1893

24.17.10.1 Configuration

BUS<m>:FLXRay:SRCType....1865
BUS<m>:FLXRay:SOURce<n>....1866
BUS<m>:FLXRay:THReshold<n>....1867
BUS<m>:FLXRay:THENable....1867
BUS<m>:FLXRay:THData....1867
BUS<m>:FLXRay:PRSingle....1867
BUS<m>:FLXRay:PRDiff....1868
BUS<m>:FLXRay:PRLogic....1868
BUS<m>:FLXRay:POLarity....1869
BUS<m>:FLXRay:BITRate....1869
BUS<m>:FLXRay:CHTYpe....1869
BUS<m>:FLXRay:SEHB....1869

BUS<m>:FLXRay:SRCType <SourceType>

Sets the type of measurement.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<SourceType> SINGle | DIFFerential | LOGic

SINGle
Used for measurements with single-ended probes or single-ended voltage measurements with differential probes on the FlexRay bus. Two thresholds have to be defined as absolute voltage levels, see `BUS<m>:FLXRay:THReshold<n>` on page 1867.

DIFFerential
Used for differential measurements on the FlexRay bus. This is the most common measurement. Two thresholds have to be defined as differential voltages.

LOGic
Used for measurements of the logic signal inside the FlexRay node, between the communication controller and the bus driver. It is possible to measure simultaneously on a data line and on the "enable" line. Each line requires its own threshold.

*RST: SINGle

BUS<m>:FLXRay:SOURce<n> <Sources>

Sets the input channel of the bus signal, or of the data and enable lines in case of a LOGic source type.

Suffix:

<m> 1..4
Selects the serial bus.

<n> 1 | 2
Selects the source: 1 = bus signal or data line, 2 = enable line

Parameters:

<Sources> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

C1W1 | C2W1 | C3W1 | C4W1
Always available

NONE
Only available for SOURce2 (enable line)

M1 | M2 | M3 | M4 | R1 | R2 | R3 | R4
Only available if the trigger source is one of the input channels but not the serial bus.

D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15
Only available if `BUS<m>:FLXRay:SOURce<n>` is set to `LOGic`.

*RST: SOURce1: C1W1, SOURce2: C2W1

BUS<m>:FLXRay:THReshold<n> <THResholds>

Sets the thresholds for the bus signal if the source type is SINGle or DIFFerential.

For LOGic source type, use `BUS<m>:FLXRay:THData` on page 1867 and `BUS<m>:FLXRay:THENable` on page 1867.

Suffix:

<m> 1..4
Selects the serial bus.

<n> 1 | 2
1 = threshold high, 2 = threshold low

Parameters:

<THResholds> Differential or absolute voltage level, depending on the source type. See `BUS<m>:FLXRay:SRCType` on page 1865.

BUS<m>:FLXRay:THENable <ThresholdEnable>

Sets the threshold for the enable line if the source type is LOGic.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<ThresholdEnable>
Range: -12 to 12
Increment: 0.1
*RST: 2.65
Default unit: V

BUS<m>:FLXRay:THData <ThresholdData>

Sets the threshold for the data line if the source type is LOGic.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<ThresholdData>
Range: -12 to 12
Increment: 0.1
*RST: 2.35
Default unit: V

BUS<m>:FLXRay:PRSingle <PresetSingleEnd>

Selects a default threshold voltage if `BUS<m>:FLXRay:SRCType` is set to `SINGle`.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<PresetSingleEnd> MV150 | MV200 | MV250 | MV300 | MANual

MV150 | MV200 | MV250 | MV300
2.5 ± 0.15 V; 2.5 ± 0.2 V; 2.5 ± 0.25 V; 2.5 ± 0.3 V, respectively

MANual
Manual setting of user-defined values with `BUS<m>:FLXRay:THReshold<n>` on page 1867.

*RST: MV150

BUS<m>:FLXRay:PRDiff <PresetDiff>

Selects a default threshold voltage if `BUS<m>:FLXRay:SRCType` is set to `DIFFerential`.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<PresetDiff> MV150 | MV200 | MV250 | MV300 | MANual

MV150 | MV200 | MV250 | MV300
±150 mV, ±200 mV, ±250 mV, ±300 mV respectively

MANual
Manual setting of user-defined values with `BUS<m>:FLXRay:THReshold<n>` on page 1867.

*RST: MV150

BUS<m>:FLXRay:PRLogic <PresetLogic>

Selects a default threshold voltage if `BUS<m>:FLXRay:SRCType` is set to `LOGic`.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<PresetLogic> V25 | V165 | V125 | V09 | V0 | MANual

V25 | V165 | V125 | V09 | V0
2.5 V (CMOS 5.0 V); 1.65 V (CMOS 3.5V), 1.25 V (CMOS 2.5V), 0.9 V (CMOS 1.8V), 0 V (ground)

MANual
Manual setting of user-defined values with `BUS<m>:FLXRay:THReshold<n>` on page 1867.

*RST: V25

BUS<m>:FLXRay:POLarity <Polarity>

Selects the wire on which the bus signal is measured in case of SINGle source type. The setting affects the digitization of the signal.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Polarity> BPLus | BMINus
*RST: BPLus

BUS<m>:FLXRay:BITRate <Bitrate>

Selects the number of transmitted bits per second.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Bitrate> M10 | M5 | M2_5
10, 5, or 2.5 Mbit/s.
The return value of 2.5 Mbit/s is M25.
*RST: M10

Example:
```
BUS:FLXRay:BITRate M2_5
BUS:FLXRay:BITRate?
M25
```

BUS<m>:FLXRay:CHTYpe <Channel>

Selects the channel on which the signal is measured. The setting is considered in the calculation of the frame CRC.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Channel> CHA | CHB
Channel A or channel B
*RST: CHA

BUS<m>:FLXRay:SEHB <SeparateHdrBts>

The command affects the decoding and its display. If ON, the leading five indicator bits of the header are decoded as five single bits. Otherwise, the indicator bits are shown as one word with word length five bit.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<SeparateHdrBts> ON | OFF
*RST: OFF

24.17.10.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:FLXRay:TYPE 1871
TRIGger<m>:FLXRay:PLPReamble 1872
TRIGger<m>:FLXRay:NUFRame 1872
TRIGger<m>:FLXRay:SYFRame 1872
TRIGger<m>:FLXRay:STFRame 1872
TRIGger<m>:FLXRay:FCONdition 1872
TRIGger<m>:FLXRay:FMIN 1873
TRIGger<m>:FLXRay:FMAX 1873
TRIGger<m>:FLXRay:PCONdition 1873
TRIGger<m>:FLXRay:PMIN 1874
TRIGger<m>:FLXRay:PMAX 1874
TRIGger<m>:FLXRay:CENable 1874
TRIGger<m>:FLXRay:CMIN 1874
TRIGger<m>:FLXRay:CMAX 1875
TRIGger<m>:FLXRay:CSTep 1875
TRIGger<m>:FLXRay:DPOPerator 1875
TRIGger<m>:FLXRay:DPOSition 1875
TRIGger<m>:FLXRay:DPTO 1876
TRIGger<m>:FLXRay:DCONdition 1876
TRIGger<m>:FLXRay:DMIN 1876
TRIGger<m>:FLXRay:DMAX 1876
TRIGger<m>:FLXRay:SYMBol 1877
TRIGger<m>:FLXRay:BSSerror 1877
TRIGger<m>:FLXRay:FESerror 1877
TRIGger<m>:FLXRay:FSSerror 1877
TRIGger<m>:FLXRay:HCRCerror 1878
TRIGger<m>:FLXRay:PCRCerror 1878

TRIGger<m>:FLXRay:TYPE <Type>

Selects the trigger type for FlexRay analysis.

Parameters:

<Type> STOF | IDDT | SYMBol | ERRC

STOF
STart Of Frame: triggers on the first rising edge after the transmission start sequence (TSS).

IDDT
IDentifier and DaTa: triggers on the decoded frame content, on header and payload data.
For all settings that are not needed for the trigger condition, make sure to set its condition to `OFF`.
Indicator bits: see `TRIGger<m>:FLXRay:NUFRame`
Frame identifier: sets the trigger to one specific frame ID or an identifier range. To set the identifier, use `TRIGger<m>:FLXRay:FCONdition`, `TRIGger<m>:FLXRay:FMIN`, and `TRIGger<m>:FLXRay:FMAX`.
Payload length: trigger on the number of words in the payload segment. To set the payload length, use `TRIGger<m>:FLXRay:PCONdition`, `TRIGger<m>:FLXRay:PMIN`, and `TRIGger<m>:FLXRay:PMAX`.
Cycle count: trigger on the number of the current FlexRay cycle. To set the cycle count, use `TRIGger<m>:FLXRay:CENable`, `TRIGger<m>:FLXRay:CMIN`, `TRIGger<m>:FLXRay:CMAX`, and `TRIGger<m>:FLXRay:CSTep`.
Data position: sets the position of the data bit pattern within the payload segment. To set the data position, use `TRIGger<m>:FLXRay:DPOPerator`, `TRIGger<m>:FLXRay:DPOSition`, and `TRIGger<m>:FLXRay:DPTO`.
Data bit pattern: sets the data bit pattern to be found in the payload segment. The starting point of the pattern is defined by the data position. To set the bit pattern, use `TRIGger<m>:FLXRay:DCONdition`, `TRIGger<m>:FLXRay:DMIN`, and `TRIGger<m>:FLXRay:DMAX`.

SYMBol
Triggers on a symbol or wakeup pattern. Set the required symbol with `TRIGger<m>:FLXRay:SYMBol`

ERRC
ERRor Condition: triggers on one or more errors that are detected in the decoded data. Use `TRIGger<m>:FLXRay:BSSerror`, `TRIGger<m>:FLXRay:FESerror`, `TRIGger<m>:FLXRay:FSSerror`, and `TRIGger<m>:FLXRay:PCRCerror`.

*RST: STOF

TRIGger<m>:FLXRay:PLPReamble <PayloadPreamble>

Triggers on the payload preamble indicator bit that indicates a Network Management Vector in the payload segment.

Parameters:

<PayloadPreamble> ONE | ZERO | DC

Bit value: 1, 0, or X (don't care)

*RST: DC

TRIGger<m>:FLXRay:NUFRame <NullFrame>

Triggers on the null frame indicator bit, a frame without usable data.

Parameters:

<NullFrame> ONE | ZERO | DC

Bit value: 1, 0, or X (don't care)

*RST: DC

TRIGger<m>:FLXRay:SYFRame <SyncFrame>

Triggers on the sync frame used for synchronization of the FlexRay system. Only sync nodes can send this frame type.

Parameters:

<SyncFrame> ONE | ZERO | DC

Bit value: 1, 0, or X (don't care)

*RST: DC

TRIGger<m>:FLXRay:STFRame <StartupFrame>

Triggers on startup frames used for startup of the network. Only specific start nodes can send this frame type.

Parameters:

<StartupFrame> ONE | ZERO | DC

Bit value: 1, 0, or X (don't care)

*RST: DC

TRIGger<m>:FLXRay:FCONdition <IdOperator>

Sets the operator to set a frame ID or a frame ID range.

Parameters:

<IdOperator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The frame ID is not relevant for the trigger condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one frame ID to be set with `TRIGger<m>:FLXRay:FMIN`.

INRange | OORange
In range, Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:FLXRay:FMIN` and `TRIGger<m>:FLXRay:FMAX`.

*RST: EQUal

TRIGger<m>:FLXRay:FMIN <IdPattern>

Specifies a frame identifier pattern - the number of the slot - or sets the the start value of an identifier range.

Parameters:

IdPattern — Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

TRIGger<m>:FLXRay:FMAX <IdPatternTo>

Sets the the end value of an identifier range if the condition `TRIGger<m>:FLXRay:FCONdition` is set to `INRange` or `OORange`.

Parameters:

IdPatternTo — Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.
FMAX must be greater or equal than FMIN, and the position of the X bits are common to FMIN and FMAX.

TRIGger<m>:FLXRay:PCONdition <PayloadLengthOp>

Sets the operator for the payload length trigger setting. You can defined an exact value, or a range.

Parameters:

<PayloadLengthOp> — OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The payload length is not relevant for the trigger condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one payload length to be set with `TRIGger<m>:FLXRay:PMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:FLXRay:PMIN` and `TRIGger<m>:FLXRay:PMAX`.

*RST: OFF

TRIGger<m>:FLXRay:PMIN <PayloadLength>

Specifies a payload length - the number of words in the payload segment - or sets the the start value of an payload length range. Information is transmitted in 2-byte words, so the number of data bytes in the payload segment is twice the payload length.

Parameters:

<PayloadLength> Range: 0 to 127
Increment: 1
*RST: 0

TRIGger<m>:FLXRay:PMAX <PayloadLengthTo>

Sets the the end value of a payload length range if the condition `TRIGger<m>:FLXRay:PCONdition` is set to `INRange` or `OORange`.

Parameters:

<PayloadLengthTo> Range: 0 to 127
Increment: 1
*RST: 0

TRIGger<m>:FLXRay:CENable <CycleCount>

Sets the operator to define a cycle count or a cycle count range.

Parameters:

<CycleCount> OFF | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF
The cycle count is not relevant for the trigger condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one cycle count number to be set with `TRIGger<m>:FLXRay:CMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:FLXRay:CMIN` and `TRIGger<m>:FLXRay:CMAX`.

*RST: OFF

TRIGger<m>:FLXRay:CMIN <CycleCountMin>

Specifies a cycle count - the number of the current FlexRay cycle - or sets the the start value of an cycle count range.

Parameters:

<CycleCountMin> Range: 0 to 63
Increment: 1
*RST: 0

TRIGger<m>:FLXRay:CMAX <CycleCountMax>

Sets the the end value of a cycle count range if the condition `TRIGger<m>:FLXRay:CENable` on page 1874 is set to `INRange` or `OORange`.

Parameters:

<CycleCountMax> Range: 0 to 63
Increment: 1
*RST: 0

TRIGger<m>:FLXRay:CSTep <CycleCountStep>

Specifies a step to trigger on each n-th cycle inside the given range. This allows for specific triggering if slot multiplexing is used.

The condition `TRIGger<m>:FLXRay:CENable` on page 1874 must be set to `INRange` or `OORange`.

Parameters:

<CycleCountStep> Range: 1 to 63
Increment: 1
*RST: 1

TRIGger<m>:FLXRay:DPOPerator <DataPosOperator>

Sets the operator for the data position. You can defined an exact position, or a position range.

Parameters:

<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe

OFF = ANY
The data position is not relevant for the trigger condition.

EQUal | GETHan
Equal, Greater or equal than. These conditions require one data position to be set with `TRIGger<m>:FLXRay:DPOSition`.

INRange = RANGe
In range: Set the minimum and maximum value of the range with `TRIGger<m>:FLXRay:DPOSition` and `TRIGger<m>:FLXRay:DPTO`.

*RST: EQUal

TRIGger<m>:FLXRay:DPOSition <DataPosition>

Sets the number of data bytes to be skipped after start of the payload segment

Parameters:

<DataPosition> Range: 0 to 255
Increment: 1
*RST: 0

TRIGger<m>:FLXRay:DPTO <DataPositionTo>

Defines the last byte of interest, if the position operator `TRIGger<m>:FLXRay:DPOPerator` defines a range.

Parameters:

<DataPositionTo> Range: 0 to 255
Increment: 1
*RST: 0

TRIGger<m>:FLXRay:DCONdition <DataOperator>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<DataOperator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The data position is not relevant for the trigger condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data position to be set with `TRIGger<m>:FLXRay:DMIN`.

INRange | OORange
In range, Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:FLXRay:DMIN` and `TRIGger<m>:FLXRay:DMAX`.

*RST: EQUal

TRIGger<m>:FLXRay:DMIN <DataPattern>

Specifies a data pattern, or sets the the start value of a data pattern range.

Parameters:

<DataPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:FLXRay:DMAX <DataPatternTo>

Sets the the end value of an data range if the operator `TRIGger<m>:FLXRay:DCONdition` is set to INRange or OORange.

Parameters:

<DataPatternTo> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:FLXRay:SYMBol <Symbol>

Triggers on a symbol or on a wakeup pattern.

Parameters:

<Symbol> CASMts | WAKeup

CASMts
Collision Avoidance Symbol / Media access Test Symbol. These symbols are identical and can be sent in the optional symbol window at the end of a communication cycle. They are used to avoid collisions during the system start.

WAKeup
The wakeup pattern is sent to activate the nodes of the system.

*RST: CASMts

TRIGger<m>:FLXRay:BSSerror <BSSError>

Triggers on error in SyteStart Sequence. The BSS is transmitted before each byte.

Parameters:

<BSSError> ON | OFF

*RST: ON

TRIGger<m>:FLXRay:FESerror <FESError>

Triggers on error in Frame End Sequence. FES indicates the end of each frame.

Parameters:

<FESError> ON | OFF

*RST: ON

TRIGger<m>:FLXRay:FSSerror <FSSError>

Triggers on Error in a Frame Start Sequence. FSS follows the Transmission Start Sequence TSS at the beginning of each frame.

Parameters:

<FSSError> ON | OFF

*RST: ON

TRIGger<m>:FLXRay:HCRCerror <CRCHeaderError>

Triggers on error in the Cyclic Redundancy Check of the header data (mainly frame ID and payload length).

Parameters:

<CRCHeaderError> ON | OFF
*RST: ON

TRIGger<m>:FLXRay:PCRCerror <CRCPayloadError>

Triggers on error in the Cyclic Redundancy Check of the payload data.

Parameters:

<CRCPayloadError> ON | OFF
*RST: ON

24.17.10.3 Decode results

To load and acivate a label list, use:

- BUS<m>:NEWList on page 1672
- BUS<m>:SYMBols on page 1672

To show the results on the screen, use the following commands:

- BUS<m>:RESult on page 1671

BUS<m>:FLXRay:FCOunt? 1878
BUS<m>:FLXRay:FRAMe<n>:STATus? 1879
BUS<m>:FLXRay:FRAMe<n>:STARt? 1879
BUS<m>:FLXRay:FRAMe<n>:STOP? 1879
BUS<m>:FLXRay:FRAMe<n>:SYMBol? 1880
BUS<m>:FLXRay:FRAMe<n>:TYPE? 1880
BUS<m>:FLXRay:FRAMe<n>:DATA? 1880
BUS<m>:FLXRay:FRAMe<n>:FLAGs? 1881
BUS<m>:FLXRay:FRAMe<n>:ADID? 1881
BUS<m>:FLXRay:FRAMe<n>:PAYLength? 1882
BUS<m>:FLXRay:FRAMe<n>:CYCount? 1882
BUS<m>:FLXRay:FRAMe<n>:CSSTate? 1882
BUS<m>:FLXRay:FRAMe<n>:CSValue? 1883
BUS<m>:FLXRay:FRAMe<n>:FCSTate? 1883
BUS<m>:FLXRay:FRAMe<n>:FCValue? 1883

BUS<m>:FLXRay:FCOunt?

Returns the number of decoded frames.

Suffix:

<m> 1..4
Selects the serial bus.

Return values:

<Count> Returns the number of decoded frames.

Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:STATus?

Returns the overall state of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameState> OK | FSS | BSS | FES | INDicator | HCRCerror | CRCerr | LENER | LENer | HCFCerror | INSufficient

OK: the frame is valid.
FSS: Frame Start Sequence after TSS is missing.
BSS: Byte Start Sequence is missing.
FES: error in the Frame End Sequence.
INDicator: Error in indicator bits.
HCRCerror: Header CRC is not valid.
CRCerr: Payload CRC is not valid.
LENER = LENer: Unexpected length of the frame.
HCFCerror: Header CRC error and frame CRC error
INSufficient: The frame is not completely contained in the acquisition. The acquired part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:STARt?
BUS<m>:FLXRay:FRAMe<n>:STOP?

Return the start time and stop time of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameStop> Time

Range: -100E+24 to 100E+24
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:SYMBol?

Returns the symbolic label of the specified frame if the label list is enabled.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the number of the frame in the current acquisition, 1...n.

Return values:

<Label> String with symbolic name of the identifier

Example: `BUS:FLXRay:FRAMe2:SYMBol?`
Response: `Temperature`

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:TYPE?

Returns the frame type of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameType> UNKNown | STATic | DYNamic | WAKE | SYMBol

STATic: frame of the static segment
DYNamic: frame of the dynamic segment
WAKE: frame contains wakeup pattern
SYMBol: frame contains a MTS or CAS symbol

*RST: STATic

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:DATA?

Returns the data of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Data> Comma-separated list of integer values (N, D1, D2,…, DN). N is the number of bytes in the frame, and D1…DN are the values of the bytes.

Example:

```
BUS:FLXRay:FRAMe4:DATA?
<-- 4,17,85,170,85
```

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:FLAGs?

Returns the value of the indicator bits at the beginning of the header segment. The five bits are read as one word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Flags> Range: 0 to 31
*RST: 0

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:ADID?

Returns the frame identifier, the number of the slot in which the frame is transmitted.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<AddressID> To set the value format, use `FORMat:BPATtern` on page 1294. The values below – range, increment and default – are decimal values.

Range: 0 to 2047
*RST: 0

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:PAYLength?

Returns the payload length, the number of data words in the payload segment. Information is transmitted in 2-byte words, so the number of data bytes in the payload segment is twice the payload length.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<PayLen> Range: 0 to 127
*RST: 0

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:CYCount?

Returns the number of the current FlexRay cycle.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<CycleCount> Range: 0 to 63
*RST: 0

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:CSSTate?

Returns the state of the cyclic redundancy check code of the header data.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<HdrCksumSt> OK | UVAL | INSufficient

OK: the CRC is valid.
UVAL: unexpected value
INSufficient: the frame is not completely contained in the acquisition. The decoded part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:CSValue?

Returns the checksum value of the header CRC.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<HdrCksumVal> To set the value format, use `FORMat:BPATtern` on page 1294. The values below – range, increment and default – are decimal values.

Range: 0 to 2047
*RST: 0

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:FCSTate?

Returns the state of the cyclic redundancy check code of the frame data.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<ChecksumState> OK | UVAL | INSufficient

OK: the CRC is valid.
UVAL: unexpected value
INSufficient: the frame is not completely contained in the acquisition. The decoded part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:FLXRay:FRAMe<n>:FCValue?

Returns the cyclic redundancy check code of the frame CRC.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<ChecksumValue> To set the value format, use `FORMat:BPATtern` on page 1294. The values below – range, increment and default – are decimal values.

Range: 0 to 16777215
*RST: 0

Usage: Query only

24.17.10.4 Search settings

SEARch:TRIGger:FLXRay:SERRor 1884
SEARch:TRIGger:FLXRay:SIDData 1885
SEARch:TRIGger:FLXRay[:SSOFrame] 1885
SEARch:TRIGger:FLXRay:SSYMbol 1885
SEARch:TRIGger:FLXRay:CENable 1885
SEARch:TRIGger:FLXRay:CMAX 1886
SEARch:TRIGger:FLXRay:CMIN 1886
SEARch:TRIGger:FLXRay:CSTep 1886
SEARch:TRIGger:FLXRay:DCONdition 1887
SEARch:TRIGger:FLXRay:DMAX 1887
SEARch:TRIGger:FLXRay:DMIN 1887
SEARch:TRIGger:FLXRay:DPOPerator 1888
SEARch:TRIGger:FLXRay:DPOSition 1888
SEARch:TRIGger:FLXRay:DPTO 1888
SEARch:TRIGger:FLXRay:FCONdition 1889
SEARch:TRIGger:FLXRay:FMAX 1889
SEARch:TRIGger:FLXRay:FMIN 1889
SEARch:TRIGger:FLXRay:NUFRame 1890
SEARch:TRIGger:FLXRay:PLPReamble 1890
SEARch:TRIGger:FLXRay:PCONdition 1890
SEARch:TRIGger:FLXRay:PMAX 1891
SEARch:TRIGger:FLXRay:PMIN 1891
SEARch:TRIGger:FLXRay:STFRame 1891
SEARch:TRIGger:FLXRay:SYFRame 1892
SEARch:TRIGger:FLXRay:SYMBol 1892
SEARch:TRIGger:FLXRay:BSSerror 1892
SEARch:TRIGger:FLXRay:FESerror 1892
SEARch:TRIGger:FLXRay:FSSerror 1893
SEARch:TRIGger:FLXRay:HCRCerror 1893
SEARch:TRIGger:FLXRay:PCRCerror 1893

SEARch:TRIGger:FLXRay:SERRor <SearchName>,<ErrorCondition>
SEARch:TRIGger:FLXRay:SERRor? <SearchName>

Enables the search for specified error or error combination.

Parameters:

<ErrorCondition> ON | OFF

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:SIDData <SearchName>,<IdentifierData>
SEARch:TRIGger:FLXRay:SIDData? <SearchName>

Enables the search for indentifier and data.

Parameters:
<IdentifierData> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay[:SSOFrame] <SearchName>,<FrameStart>
SEARch:TRIGger:FLXRay[:SSOFrame]? <SearchName>

Enables the search for a start of frame.

Parameters:
<FrameStart> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:SSYMbol <SearchName>,<CheckSymbol>
SEARch:TRIGger:FLXRay:SSYMbol? <SearchName>

Enables the search for specified symbol.

Parameters:
<CheckSymbol> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:CENable <SearchName>,<CycleCount>
SEARch:TRIGger:FLXRay:CENable? <SearchName>

Sets the operator to define a cycle count or a cycle count range.

Parameters:
<CycleCount> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The cycle count is not relevant for the search condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one cycle count number to be set with `SEARch:TRIGger:FLXRay:CMIN`.

INRange | OORange
In range, Out of range: Set the minimum and maximum value of the range with `SEARch:TRIGger:FLXRay:CMIN` and `SEARch:TRIGger:FLXRay:CMAX`.

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:CMAX <SearchName>,<CycleCountMax>
SEARch:TRIGger:FLXRay:CMAX? <SearchName>

Sets the the end value of a cycle count range if the condition `SEARch:TRIGger:FLXRay:CENable` is set to `INRange` or `OORange`.

Parameters:
<CycleCountMax> Range: 0 to 63
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:CMIN <SearchName>,<CycleCountMin>
SEARch:TRIGger:FLXRay:CMIN? <SearchName>

Specifies a cycle count - the number of the current FlexRay cycle - or sets the the start value of an cycle count range.

Parameters:
<CycleCountMin> Range: 0 to 63
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:CSTep <SearchName>,<CycleCountStep>
SEARch:TRIGger:FLXRay:CSTep? <SearchName>

Specifies a step to search for each n-th cycle inside the given range. This allows for a specific search if slot multiplexing is used.

The condition `SEARch:TRIGger:FLXRay:CENable` must be set to `INRange` or `OORange`.

Parameters:

<CycleCountStep> Range: 1 to 63
Increment: 1
*RST: 1

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:FLXRay:DCONdition? <SearchName>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<DataOperator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The data pattern is not relevant for the search condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with `SEARch:TRIGger:FLXRay:DMIN`.

INRange | OORange
In range, Out of range: Set the minimum and maximum value of the range with `SEARch:TRIGger:FLXRay:DMIN` and `SEARch:TRIGger:FLXRay:DMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:DMAX <SearchName>,<DataPatternTo>
SEARch:TRIGger:FLXRay:DMAX? <SearchName>

Sets the the end value of an data range if `SEARch:TRIGger:FLXRay:DCONdition` is set to `INRange` or `OORange`.

Parameters:

<DataPatternTo> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:DMIN <SearchName>,<DataPattern>
SEARch:TRIGger:FLXRay:DMIN? <SearchName>

Specifies a data pattern, or sets the the start value of a data pattern range.

Parameters:
<DataPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:DPOPerator <SearchName>,<DataPosOperator>
SEARch:TRIGger:FLXRay:DPOPerator? <SearchName>

Sets the operator for the data position. You can defined an exact position, or a position range.

Parameters:
<DataPosOperator> ANY | OFF | EQUal | GETHan | INRange | RANGe

OFF = ANY
The data position is not relevant for the search condition.

EQUal | GETHan
Equal, Greater or equal than. These conditions require one data position to be set with `SEARch:TRIGger:FLXRay:DPOSition`

INRange = RANGe
In range: Set the minimum and maximum value of the range with `SEARch:TRIGger:FLXRay:DPOSition` and `SEARch:TRIGger:FLXRay:DPTO`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:DPOSition <SearchName>,<DataPosition>
SEARch:TRIGger:FLXRay:DPOSition? <SearchName>

Sets the number of data bytes to be skipped after start of the payload segment.

Parameters:
<DataPosition> Range: 0 to 255
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:DPTO <SearchName>,<DataPositionTo>
SEARch:TRIGger:FLXRay:DPTO? <SearchName>

Defines the last byte of interest, if the position operator `SEARch:TRIGger:FLXRay:DPOPerator` defines a range.

Parameters:

<DataPositionTo> Range: 0 to 255
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:FCONdition <SearchName>,<IdOperator>
SEARch:TRIGger:FLXRay:FCONdition? <SearchName>

Sets the operator to set a frame ID or a frame ID range.

Parameters:

<IdOperator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The frame ID is not relevant for the search condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one frame ID to be set with `SEARch:TRIGger:FLXRay:FMIN`.

INRange | OORange
In range, Out of range: Set the minimum and maximum value of the range with `SEARch:TRIGger:FLXRay:FMIN` and `SEARch:TRIGger:FLXRay:FMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:FMAX <SearchName>,<IdPatternTo>
SEARch:TRIGger:FLXRay:FMAX? <SearchName>

Sets the the end value of an identifier range if the condition `SEARch:TRIGger:FLXRay:FCONdition` is set to `INRange` or `OORange`.

Parameters:

<IdPatternTo> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:FMIN <SearchName>,<IdPattern>
SEARch:TRIGger:FLXRay:FMIN? <SearchName>

Specifies a frame identifier pattern - the number of the slot - or sets the the start value of an identifier range.

Parameters:
<IdPattern> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:NUFRame <SearchName>,<NullFrame>
SEARch:TRIGger:FLXRay:NUFRame? <SearchName>

Searches for the null frame indicator bit, a frame without usable data.

Parameters:
<NullFrame> ONE | ZERO | DC

Bit value: 1, 0, or X (don't care)

*RST: DC

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:PLPReamble <SearchName>,<PayloadPreamble>
SEARch:TRIGger:FLXRay:PLPReamble? <SearchName>

Searches for the payload preamble indicator bit that indicates a Network Management Vector in the payload segment.

Parameters:
<PayloadPreamble> ONE | ZERO | DC

Bit value: 1, 0, or X (don't care)

*RST: DC

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:PCONdition <SearchName>,<PayloadLengthOp>
SEARch:TRIGger:FLXRay:PCONdition? <SearchName>

Sets the operator for the payload length search setting. You can defined an exact value, or a range.

Parameters:
<PayloadLengthOp> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The payload length is not relevant for the search condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one payload length to be set with `SEARch:TRIGger:FLXRay:PMIN`.

INRange | OORange
In range, Out of range: Set the minimum and maximum value of the range with `SEARch:TRIGger:FLXRay:PMIN` and `SEARch:TRIGger:FLXRay:PMAX`.

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:PMAX <SearchName>,<PayloadLengthTo>
SEARch:TRIGger:FLXRay:PMAX? <SearchName>

Sets the the end value of a payload length range if the condition `SEARch:TRIGger:FLXRay:PCONdition` is set to `INRange` or `OORange`.

Parameters:
<PayloadLengthTo> Range: 0 to 127
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:PMIN <SearchName>,<PayloadLength>
SEARch:TRIGger:FLXRay:PMIN? <SearchName>

Specifies a payload length - the number of words in the payload segment - or sets the the start value of an payload length range. Information is transmitted in 2-byte words, so the number of data bytes in the payload segment is twice the payload length.

Parameters:
<PayloadLength> Range: 0 to 127
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:STFRame <SearchName>,<StartupFrame>
SEARch:TRIGger:FLXRay:STFRame? <SearchName>

Searches for startup frames used for startup of the network. Only specific start nodes can send this frame type.

Parameters:
<StartupFrame> ONE | ZERO | DC
Bit value: 1, 0, or X (don't care)
*RST: DC

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:SYFRame <SearchName>,<SyncFrame>
SEARch:TRIGger:FLXRay:SYFRame? <SearchName>

Searches for the sync frame used for synchronization of the FlexRay system. Only sync nodes can send this frame type.

Parameters:
<SyncFrame> ONE | ZERO | DC
Bit value: 1, 0, or X (don't care)
*RST: DC

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:SYMBol <SearchName>,<Symbol>
SEARch:TRIGger:FLXRay:SYMBol? <SearchName>

Searches for a symbol or for a wakeup pattern.

Parameters:
<Symbol> CASMts | WAKeup

CASMts
Collision Avoidance Symbol / Media access Test Symbol. These symbols are identical and can be sent in the optional symbol window at the end of a communication cycle. They are used to avoid collisions during the system start.

WAKeup
The wakeup pattern is sent to activate the nodes of the system.

*RST: CASMts

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:BSSerror <SearchName>,<BSSError>
SEARch:TRIGger:FLXRay:BSSerror? <SearchName>

Searches for error in SyteStart Sequence. The BSS is transmitted before each byte.

Parameters:
<BSSError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:FESerror <SearchName>,<FESError>
SEARch:TRIGger:FLXRay:FESerror? <SearchName>

Searches for error in Frame End Sequence. FES indicates the end of each frame.

Parameters:
<FESError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:FSSerror <SearchName>,<FSSError>
SEARch:TRIGger:FLXRay:FSSerror? <SearchName>

Searches for an error in a Frame Start Sequence(FSS). FSS follows the Transmission Start Sequence (TSS) at the beginning of each frame.

Parameters:
<FSSError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:HCRCerror <SearchName>,<CRCHeaderError>
SEARch:TRIGger:FLXRay:HCRCerror? <SearchName>

Searches for an error in the Cyclic Redundancy Check of the header data (mainly frame ID and payload length).

Parameters:
<CRCHeaderError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:FLXRay:PCRCerror <SearchName>,<CRCPayloadError>
SEARch:TRIGger:FLXRay:PCRCerror? <SearchName>

Searches for error in the Cyclic Redundancy Check of the payload data.

Parameters:
<CRCPayloadError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

24.17.10.5 Search results

The search on decoded FlexRay data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 24.17.10.3, "Decode results", on page 1878.

SEARch:RESult:FLXRay:FCOunt? 1894
SEARch:RESult:FLXRay:FRAMe<m>:ADID? 1894
SEARch:RESult:FLXRay:FRAMe<m>:CSSTate? 1894
SEARch:RESult:FLXRay:FRAMe<m>:CSValue? 1895
SEARch:RESult:FLXRay:FRAMe<m>:CYCount? 1895
SEARch:RESult:FLXRay:FRAMe<m>:DATA? 1895
SEARch:RESult:FLXRay:FRAMe<m>:FCSTate? 1896
SEARch:RESult:FLXRay:FRAMe<m>:FCValue? 1896
SEARch:RESult:FLXRay:FRAMe<m>:FLAGs? 1896
SEARch:RESult:FLXRay:FRAMe<m>:PAYLength? 1897
SEARch:RESult:FLXRay:FRAMe<m>:STATus? 1897
SEARch:RESult:FLXRay:FRAMe<m>:STARt? 1898
SEARch:RESult:FLXRay:FRAMe<m>:STOP? 1898
SEARch:RESult:FLXRay:FRAMe<m>:TYPE? 1898
SEARch:RESult:FLXRay:FRAMe<m>:SYMBol? 1898

SEARch:RESult:FLXRay:FCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:

<Count>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:ADID? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<AddressID>	Range:	0 to 2047
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:CSSTate? <SearchName>

Returns the status of the header checksum.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<HeadCksumState> OK | STERror | SPERror | PRERror | UVAL | INSufficient

STERror: StarT ERror, incorrect start bit
SPERror: StoP ERror, incorrect stop bit
PRERror:PaRity ERror, incorrect parity bit.
UVAL: unexpected value

*RST: OK

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:CSValue? <SearchName>

Returns the value of the header checksum.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<HeadCksumValue> Range: 0 to 2047
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:CYCount? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<CycleCount> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:DATA? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:FCSTate? <SearchName>

Returns the status of the frame checksum.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<ChecksumState> OK | STERror | SPERror | PRERror | UVAL | INSufficient

STERror: StarT ERror, incorrect start bit
SPERror: StoP ERror, incorrect stop bit
PRERror:PaRity ERror, incorrect parity bit.
UVAL: unexpected value

*RST: OK

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:FCValue? <SearchName>

Returns the value of the frame checksum.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<ChecksumValue> Range: 0 to 16777215
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:FLAGs? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Flags> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:PAYLength? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<PayLen> Range: 0 to 127
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:STATus? <SearchName>

Returns the overall state of the selected frame.

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<State> OK | FSS | BSS | FES | INDicator | HCRCerror | CRCerr | LENER | LENer | HCFCerror | INSufficient

OK: the frame is valid.
FSS: Frame Start Sequence after TSS is missing.
BSS: Byte Start Sequence is missing.
FES: error in the Frame End Sequence.
INDicator: Error in indicator bits.
HCRCerror: Header CRC is not valid.
CRCerr: Payload CRC is not valid.
LENER = LENer: Unexpected length of the frame.
HCFCerror: Header CRC error and frame CRC error
INSufficient: The frame is not completely contained in the acquisition. The acquired part of the frame is valid.

*RST: OK

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:STOP? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:TYPE? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Type> UNKNown | STATic | DYNamic | WAKE | SYMBol
*RST: STATic

Usage: Query only

SEARch:RESult:FLXRay:FRAMe<m>:SYMBol? <SearchName>

Returns the symbolic label of the specified frame if the label list is enabled.

Suffix:
<m> *
Selects the number of the frame in the current acquisition, 1...n.

Query parameters:
<SearchName> String parameter that contains the search definition name

Return values:
<Translation> Symbolic label (string)

Usage: Query only

24.17.11 SENT (option R&S RTO6-K520)

- Configuration....1899
- Trigger....1902
- Decode results....1909
- SENT search settings....1916
- SENT search results....1924

24.17.11.1 Configuration

BUS<m>:SENT:DATA:SOURce....1899
BUS<m>:SENT:DATA:THReshold....1899
BUS<m>:SENT:TECHnology....1900
BUS<m>:SENT:CLKPeriod....1900
BUS<m>:SENT:CLKTolerance....1900
BUS<m>:SENT:DNIBbles....1901
BUS<m>:SENT:SFORmat....1901
BUS<m>:SENT:CRCVersion....1901
BUS<m>:SENT:CRCMethod....1901
BUS<m>:SENT:PPULse....1902
BUS<m>:SENT:PPFLength....1902

BUS<m>:SENT:DATA:SOURce <DataSource>

Selects the source of the data line.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<DataSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4
*RST: C1W1

Usage: Asynchronous command

BUS<m>:SENT:DATA:THReshold <Threshold>

Sets a user-defined threshold value. Alternatively, you can set the threshold according to the signal technology `BUS<m>:SENT:TECHnology`.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Threshold> Range: -12 to 12
Increment: 0.1
*RST: 1.65
Default unit: V

BUS<m>:SENT:TECHnology <Technology>

Selects the threshold voltage.

Suffix:
<m> 1..4
Selects the serial data bus.

Parameters:
<Technology> V25 | MANual

V25
The threshold value is 2.5 V, according to CMOS technology.

MANual
Sets the threshold to the value set with `BUS<m>:SENT:DATA:THReshold`.

*RST: V25

BUS<m>:SENT:CLKPeriod <ClockPeriod>

Sets the nominal clock period (clock tick).

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ClockPeriod> Range: 1E-6 to 100E-6
Increment: 1E-6
*RST: 6E-6
Default unit: s

BUS<m>:SENT:CLKTolerance <ClockTolerance>

Sets a tolerated deviation of the clock signal.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:

<ClockTolerance> Range: 0 to 25
Increment: 1
*RST: 20
Default unit: %

BUS<m>:SENT:DNIBbles <DataNibbles>

Sets the number of data nibbles for a transmission sequence.

Suffix:

<m> 1..4
Selects the serial data bus.

Parameters:

<DataNibbles> Range: 1 to 6
Increment: 1
*RST: 6

BUS<m>:SENT:SFORmat <Format>

Selects the serial message format.

Suffix:

<m> 1..4

Parameters:

<Format> SHORt | ENHanced | NONE

Short serial message, Enhanced serial message, none = single transmission sequence.

*RST: NONE

BUS<m>:SENT:CRCVersion <CRCVersion>

Selects the calculaction method for the cyclic redundancy check (CRC).

Suffix:

<m> 1..4
Selects the serial data bus.

Parameters:

<CRCVersion> LEGA | V2010

Legacy: method used up to 2010
V2010: current method

*RST: V2010

BUS<m>:SENT:CRCMethod <CRCCalculation>

Selects the calculation method for the CRC checksum.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<CRCCalculation> SAEJ | TLE

SAEJ: according to the standard
TLE: according to the computing method for TLE_4998X sensors.

*RST: SAEJ

BUS<m>:SENT:PPULse <PausePulse>

Determines if a pause pulse is part of the SENT transmission sequence.

Suffix:
<m> 1..4
Selects the serial data bus.

Parameters:
<PausePulse> NPP | PP | PPFL

NPP
Transmits the SENT message without pause pulse.

PP
Transmits the message with a fixed pulse length, automatically calculated.

PPFL
Transmits the pause pulse with a user-defined frame length to obtain a transmission sequence with constant length.

*RST: PPFL

BUS<m>:SENT:PPFLength <FrameLength>

Defines a constant transmission sequence length. To select the fixed sequence length, set `BUS:SENT:PPUL PPFL`.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<FrameLength> Range: 104 to 922
Increment: 1
*RST: 256

24.17.11.2 Trigger

Event in a trigger sequence: 1 = A-event only

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:SENT:TYPE....1903
TRIGger<m>:SENT:TTYPe....1904
TRIGger<m>:SENT:STATus....1904
TRIGger<m>:SENT:TDCN....1905
TRIGger<m>:SENT:TDMN....1905
TRIGger<m>:SENT:TDMX....1905
TRIGger<m>:SENT:STYPe....1905
TRIGger<m>:SENT:SIDType....1906
TRIGger<m>:SENT:SICN....1906
TRIGger<m>:SENT:SIMN....1906
TRIGger<m>:SENT:SIMX....1906
TRIGger<m>:SENT:SDCN....1907
TRIGger<m>:SENT:SDMN....1907
TRIGger<m>:SENT:SDMX....1907
TRIGger<m>:SENT:FORMerror....1907
TRIGger<m>:SENT:PULSeerror....1908
TRIGger<m>:SENT:PPERioderror....1908
TRIGger<m>:SENT:CRCerror....1908
TRIGger<m>:SENT:IRFLength....1909

TRIGger<m>:SENT:TYPE <Type>

Selects the trigger event for the SENT transmission type.

Parameters:

<Type> CALI | TSEQ | SMSG | ERRC

CALI
CALIbration: triggers on the falling edge of the calibration/ synchronization pulse.

TSEQ
Transmission SEQuence: triggers either on the falling edge of the status nibble, or on the last data nibble.
To set the transmission sequence conditions, use `TRIGger<m>:SENT:TTYPe` and `TRIGger<m>:SENT:STATus`.
To set the data condition, use `TRIGger<m>:SENT:TDCN`, `BUS<m>:SENT:DNIBbles`, `TRIGger<m>:SENT:TDMN` and `TRIGger<m>:SENT:TDMX`.

SMSG
Serial Message: combination of identifier and data conditions.
To select the sequence condition, use `TRIGger<m>:SENT:STYPe`.
To select the message ID format for an enhanced serial message, use `TRIGger<m>:SENT:SIDType`.
To set the identifier condition, use `TRIGger<m>:SENT:SICN`, `TRIGger<m>:SENT:SIMN` and `TRIGger<m>:SENT:SIMX`.
To set the data condition, use `TRIGger<m>:SENT:SDCN`, `TRIGger<m>:SENT:SDMN` and `TRIGger<m>:SENT:SDMX`.

ERRC
ERRor Condition: triggers on an error event.
Define the error types with `TRIGger<m>:SENT:PULSeerror`, `TRIGger<m>:SENT:PPERioderror` or `TRIGger<m>:SENT:CRCerror`.

*RST: CALI

TRIGger<m>:SENT:TTYPe <TSFieldType>

Selects the trigger sequence type for `TRIGger<m>:SENT:TYPE TSEQ` (transmission sequence).

Parameters:
<TSFieldType> STAT | STDA

STAT
Triggers on the status nibble.

STDA
Triggers at the end of the combination of status and data nibble(s).
Define the data conditions with `TRIGger<m>:SENT:STATus`, `TRIGger<m>:SENT:TDCN`, `BUS<m>:SENT:DNIBbles`, `TRIGger<m>:SENT:TDMN` and `TRIGger<m>:SENT:TDMX`

*RST: STAT

TRIGger<m>:SENT:STATus <StatusBits>

Sets the status nibble data.

Parameters:
<StatusBits> Numeric or string pattern, Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:SENT:TDCN <TSDataOperator>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<TSDataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one data pattern to be set with `TRIGger<m>:SENT:TDMN`.

INRange | OORange
In range / Out of range. To define the range set the minimum and maximum values with `TRIGger<m>:SENT:TDMN` and `TRIGger<m>:SENT:TDMX`.

*RST: EQUal

TRIGger<m>:SENT:TDMN <TSDataPattern>

Specifies a data pattern, or sets the start value of a data pattern range.

Parameters:

<TSDataPattern> Numeric or string pattern, Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:SENT:TDMX <TSDataPatternTo>

Sets the end value of an identifier range for `TRIGger:SENT:TDCN INRange or OORange`.

Parameters:

<TSDataPatternTo> Numeric or string pattern, Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:SENT:STYPe <SMFieldType>

Selects the trigger sequence type for `TRIGger:SENT:TYPE SSMSg or ESMSg` (serial message).

Parameters:

<SMFieldType> ID | IDDT

ID
Triggers on the identifier.
To set the identifier condition for a serial message, use `TRIGger<m>:SENT:SICN`, `TRIGger<m>:SENT:SIMN` and `TRIGger<m>:SENT:SIMX`.

IDDT
Triggers at the end of the combination of identifier and data.
To set the identifier condition, use the commands shown above.
To set the data condition, use `TRIGger<m>:SENT:SDCN`, `TRIGger<m>:SENT:SDMN` and `TRIGger<m>:SENT:SDMX`.

*RST: ID

TRIGger<m>:SENT:SIDType <SMIDType>

Sets the message ID format (4 bit or 8 bit) of the enhanced serial message.

Parameters:

<SMIDType> B4 | B8

*RST: B4

TRIGger<m>:SENT:SICN <SMIDOperator>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<SMIDOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one data pattern to be set with `TRIGger<m>:SENT:SIMN`.

INRange | OORange
In range / Out of range. To define the range set the minimum and maximum values with `TRIGger<m>:SENT:SIMN` and `TRIGger<m>:SENT:SIMX`.

*RST: EQUal

TRIGger<m>:SENT:SIMN <SMIDPattern>

Specifies a message identifier pattern, or sets the start value of an identifier range.

Parameters:

<SMIDPattern> Numeric or string pattern, Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:SENT:SIMX <SMIDPatternTo>

Sets the end value of an identifier range for `TRIGger<m>:SENT:SICN INRange or OORange`.

Parameters:

<SMIDPatternTo> Numeric or string pattern, Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:SENT:SDCN <SMDataOperator>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<SMDataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one data pattern to be set with `TRIGger<m>:SENT:SDMN`.

INRange | OORange
In range / Out of range. To define the range set the minimum and maximum values with `TRIGger<m>:SENT:SDMN` and `TRIGger<m>:SENT:SDMX`.

*RST: EQUal

TRIGger<m>:SENT:SDMN <SMDataPattern>

Specifies a data pattern, or sets the start value of a data pattern range.

Parameters:

<SMDataPattern> Numeric or string pattern, Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:SENT:SDMX <SMDataPatternTo>

Sets the end value of an identifier range for `TRIGger<m>:SENT:SCONdition INRange or OORange`.

Parameters:

<SMDataPatternTo> Numeric or string pattern, Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:SENT:FORMerror <FormError>

Triggers on format errors in serial messages.

A form error occurs when at least one of the transmission sequences that form a serial message has an error.

To trigger on an error event, select the corresponding trigger type with `TRIGger<m>:SENT:TYPE ERRC`.

Parameters:

<FormError> ON | OFF

*RST: OFF

TRIGger<m>:SENT:PULSeerror <CalibPulseError>

Trigger son calibration pulse erors in transmission sequences.

An error occurs when

- the duration of the "Calibration/Sync" pulse (in ticks) is less than 56*(1-clock tolerance) or more than 56*(1+clock tolerance)
- the "Calibration/Sync" pulse duration of frame (n-1) varies by more than 1.5625% from the "Calibration/Sync" pulse duration of frame (n)

To trigger on an error event, select the correspondig trigger type with `TRIGger<m>:SENT:TYPE ERRC`.

Parameters:

<CalibPulseError> ON | OFF

*RST: OFF

TRIGger<m>:SENT:PPERioderror <PulsePeriodErr>

Triggers on pulse period errors.

An error occurs when a nibble has any of the following:

- number of ticks at low is less than 4 ticks.
- nibble value < 0 (less than 12 ticks) or > 15 (more than 27 ticks).

To trigger on an error event, select the correspondig trigger type with `TRIGger<m>:SENT:TYPE ERRC`.

Parameters:

<PulsePeriodErr> ON | OFF

*RST: OFF

TRIGger<m>:SENT:CRCerror <CRCError>

Triggers on CRC errors in both, the transmission sequences and serial messages.

A CRC error occurs when the CRC calculated by the receiver differs from the received value in the CRC sequence. The CRC length is 4 bits for transmission sequences and short serial messages, and 6 bit of enhanced serial messages.

To trigger on an error event, select the correspondig trigger type with `TRIGger<m>:SENT:TYPE ERRC`.

Parameters:
<CRCError> ON | OFF
*RST: ON

TRIGger<m>:SENT:IRFLength <IrregularFrmLen>

Triggers on frame length errors in transmission sequences when pause pulse for constant frame length is set, see `BUS<m>:SENT:PPULse PPFL`.

An error occurs when the total length of the transmission sequence (including pause pulse) does not match the frame length setting, see `BUS<m>:SENT:PPFLength` on page 1902.

To trigger on an error event, select the correspondig trigger type with `TRIGger<m>:SENT:TYPE ERRC`.

Parameters:
<IrregularFrmLen> ON | OFF
*RST: OFF

24.17.11.3 Decode results

To load and acivate a label list, use:

- `BUS<m>:NEWList` on page 1672
- `BUS<m>:SYMBols` on page 1672

BUS<m>:SENT:FCOunt?....1909
BUS<m>:SENT:FRAMe<n>:STATus?....1910
BUS<m>:SENT:FRAMe<n>:STARt?....1910
BUS<m>:SENT:FRAMe<n>:STOP?....1910
BUS<m>:SENT:FRAMe<n>:CSValue?....1911
BUS<m>:SENT:FRAMe<n>:DATA?....1911
BUS<m>:SENT:FRAMe<n>:IDTYpe?....1911
BUS<m>:SENT:FRAMe<n>:IDValue?....1912
BUS<m>:SENT:FRAMe<n>:NIBBle<o>:STATe?....1912
BUS<m>:SENT:FRAMe<n>:NIBBle<o>:VALue?....1913
BUS<m>:SENT:FRAMe<n>:PAPTicks?....1913
BUS<m>:SENT:FRAMe<n>:SCOM?....1913
BUS<m>:SENT:FRAMe<n>:SDATa?....1914
BUS<m>:SENT:FRAMe<n>:SDEXport?....1914
BUS<m>:SENT:FRAMe<n>:SYMBol?....1914
BUS<m>:SENT:FRAMe<n>:SYNCduration?....1915
BUS<m>:SENT:FRAMe<n>:TYPE?....1915
BUS<m>:SENT:RDSL....1916
BUS<m>:SENT:FRAMe<n>:BITRate?....1916

BUS<m>:SENT:FCOunt?

Returns the number of decoded frames of the acquisition.

Suffix:

<m> 1..4
Selects the serial data bus.

Return values:

<Count> Total number of decoded frames.

Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:SENT:FRAMe<n>:STATus?

Returns the overall state of the selected frame.

Suffix:

<m> 1..4
Selects the serial data bus.

<n> *
Selects the frame.

Return values:

<FrameState> OK | SYNC | PULSe | CRC | IRFL | FORM | INSufficient

OK: the frame is valid.
SYNC: Synchronization error occurred.
PULse: Pulse error occurred.
CRC: Cyclic redundancy check failed.
IRFL: Irregular frame length error occurred.
FORM: Format error occurred.
INSufficient: The frame is not completely contained in the acquisition. The acquired part of the frame is valid.

*RST: OK

Usage: Query only

BUS<m>:SENT:FRAMe<n>:STARt?
BUS<m>:SENT:FRAMe<n>:STOP?

Returns the start time and stop time of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Start>, <Stop>

Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:SENT:FRAMe<n>:CSValue?

Returns the CRC sequence value of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<ChecksumValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: 0 to 63
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:SENT:FRAMe<n>:DATA?

Returns the data of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Parameters:

<Data> Comma-separated sequence of integer values (N, D1, D2,…, DN). N is the number of nibbles in the frame, and D1…DN are the values of the nibbles.

Example:

```
BUS:SENT:FRAMe4:DATA?
<-- 4,3,15,11,9
```

Usage: Query only

BUS<m>:SENT:FRAMe<n>:IDTYpe?

Returns the identifier type of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<IdentifierType> B4 | B8

B4: standard format, 4 bit
B8: extended format, 8 bit

*RST: B4

Usage: Query only

BUS<m>:SENT:FRAMe<n>:IDValue?

Returns the identifier value of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<IdentifierValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:SENT:FRAMe<n>:NIBBle<o>:STATe?

Returns the state of the specified nibble.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the nibble number.

Return values:

<State> OK | UNDF

UNDF: Undefined

*RST: OK

Usage: Query only

BUS<m>:SENT:FRAMe<n>:NIBBle<o>:VALue?

Returns the value of the specified nibble.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the nibble number.

Return values:

<Value> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: 0 to 15
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:SENT:FRAMe<n>:PAPTicks?

Returns the number of the pulse pause clock ticks.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<PausePulseTicks> Range: 12 to 768
Increment: 1
*RST: 12

Usage: Query only

BUS<m>:SENT:FRAMe<n>:SCOM?

Returns the value of the status/communication pulse.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<StatusComm> Range: 0 to 15
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:SENT:FRAMe<n>:SDATa?

Returns the symbolic data of the frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<SymbolicData> Comma-separated list of values. The first value is the number of bytes, followed by the decoded data bytes.
To set the value format, use `FORMat:BPATtern`.

Usage: Query only

BUS<m>:SENT:FRAMe<n>:SDEXport?

Returns the symbolic data of the frame in export format.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<SymbolicData>

Usage: Query only

BUS<m>:SENT:FRAMe<n>:SYMBol?

Returns the symbolic label of the specified frame if the label list is enabled.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the number of the frame in the current acquisition, 1...n.

Return values:

<Translation> String with symbolic label of the identifier.

Example: `BUS:SENT:FRAMe:SYMBol?`
Response: `Air Temperature`

Usage: Query only

BUS<m>:SENT:FRAMe<n>:SYNCduration?

Returns the time of the synchronization pulse.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<SyncDuration> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:SENT:FRAMe<n>:TYPE?

Returns the type of SENT message.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameType> TRSQ | SMSG | EMSG

Transmission sequence, short serial message or enhanced serial message.

*RST: TRSQ

Usage: Query only

BUS<m>:SENT:RDSL <RessDispSel>

Selects the results to be displayed.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<RessDispSel> TRSQ | SMSG | ALL
Transmission sequence, serial messages or all.
*RST: ALL

BUS<m>:SENT:FRAMe<n>:BITRate?

Returns the bit rate of the selected frame.

Suffix:
<m> 1..4
<n> *

Return values:
<PrimaryBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

24.17.11.4 SENT search settings

In search setup commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name. The commands are similar to SENT trigger commands.

SEARch:TRIGger:SENT:CALibration....1917
SEARch:TRIGger:SENT:TRANsmission....1917
SEARch:TRIGger:SENT:SMSG....1917
SEARch:TRIGger:SENT:ERRor....1918
SEARch:TRIGger:SENT:TTYPe....1918
SEARch:TRIGger:SENT:STATus....1919
SEARch:TRIGger:SENT:TDCN....1919
SEARch:TRIGger:SENT:TDMN....1919
SEARch:TRIGger:SENT:TDMX....1920
SEARch:TRIGger:SENT:STYPe....1920
SEARch:TRIGger:SENT:SIDType....1920
SEARch:TRIGger:SENT:SICN....1921
SEARch:TRIGger:SENT:SIMN....1921
SEARch:TRIGger:SENT:SIMX....1921
SEARch:TRIGger:SENT:SDCN....1922
SEARch:TRIGger:SENT:SDMN....1922

SEARch:TRIGger:SENT:SDMX 1922
SEARch:TRIGger:SENT:PULSeerror 1923
SEARch:TRIGger:SENT:PPERioderror 1923
SEARch:TRIGger:SENT:IRFLength 1923
SEARch:TRIGger:SENT:FORMerror 1923
SEARch:TRIGger:SENT:CRCerror 1924

SEARch:TRIGger:SENT:CALibration <SearchName>,<CalSyncPulse>
SEARch:TRIGger:SENT:CALibration? <SearchName>

Enables the search for the Calibration/Synchronization pulse.

Parameters:
<CalSyncPulse> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:TRANsmission <SearchName>,<TransmSequence>
SEARch:TRIGger:SENT:TRANsmission? <SearchName>

Enables the search for a transmission sequence, which is combination of status and data conditions.

To set the transmission sequence conditions, use `SEARch:TRIGger:SENT:TTYPe` and `SEARch:TRIGger:SENT:STATus`.

To set the data condition, use `SEARch:TRIGger:SENT:TDCN`, `TRIGger<m>:SENT:TDCN`, `SEARch:TRIGger:SENT:TDMN` and `SEARch:TRIGger:SENT:TDMX`.

Parameters:
<TransmSequence> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:SMSG <SearchName>,<SerialMessage>
SEARch:TRIGger:SENT:SMSG? <SearchName>

Enables the search in a serial message, which is combination of identifier and data conditions.

To select the sequence condition, use `SEARch:TRIGger:SENT:STYPe`.

To set the identifier condition for the serial message, use `SEARch:TRIGger:SENT:SICN`, `SEARch:TRIGger:SENT:SIMN` and `SEARch:TRIGger:SENT:SIMX`.

To set the data condition, use `SEARch:TRIGger:SENT:SDCN`, `SEARch:TRIGger:SENT:SDMN` and `SEARch:TRIGger:SENT:SDMX`.

Parameters:
<SerialMessage> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:ERRor <SearchName>,<ErrorCondition>
SEARch:TRIGger:SENT:ERRor? <SearchName>

Enables the search for specified errors.

Define the error types with `SEARch:TRIGger:SENT:PULSeerror`, `SEARch:TRIGger:SENT:PPERioderror`, `SEARch:TRIGger:SENT:FORMerror` on page 1923 and `SEARch:TRIGger:SENT:CRCerror`.

Parameters:
<ErrorCondition> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:TTYPe <SearchName>,<TSFieldType>
SEARch:TRIGger:SENT:TTYPe? <SearchName>

Selects the SENT transmission sequence to be searched for.

To enable the search for the transmission sequence, use `SEARch:TRIGger:SENT:TRANsmission`.

Parameters:
<TSFieldType> STAT | STDA

STAT
Searches on the status nibble.

STDA
Searches for the end of the combination of status and data nibble(s).
Define the data conditions with `SEARch:TRIGger:SENT:STATus`, `SEARch:TRIGger:SENT:TDCN`, `BUS<m>:SENT:DNIBbles`, `SEARch:TRIGger:SENT:TDMN` and `SEARch:TRIGger:SENT:TDMX`.

*RST: STAT

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:STATus <SearchName>,<StatusBits>
SEARch:TRIGger:SENT:STATus? <SearchName>

Sets the status nibble data.

Parameters:

<StatusBits> Numeric or string pattern, Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:TDCN <SearchName>,<TSDataOperator>
SEARch:TRIGger:SENT:TDCN? <SearchName>

Sets the operator for a specific data pattern or a data pattern range.

Parameters:

<TSDataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one data pattern to be set with `SEARch:TRIGger:SENT:TDMN`.

INRange | OORange
In range / Out of range. To define the range set the minimum and maximum values with `SEARch:TRIGger:SENT:TDMN` and `SEARch:TRIGger:SENT:TDMX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:TDMN <SearchName>,<TSDataPattern>
SEARch:TRIGger:SENT:TDMN? <SearchName>

Specifies a data pattern, or sets the start value of a data pattern range.

Parameters:

<TSDataPattern> Numeric or string pattern, Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:TDMX <SearchName>,<TSDataPatternTo>
SEARch:TRIGger:SENT:TDMX? <SearchName>

Sets the end value of an identifier range for `SEARch:TRIGger:SENT:DCONdition INRange or OORange`.

Parameters:

<TSDataPatternTo> Numeric or string pattern, Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:STYPe <SearchName>,<SMFieldType>
SEARch:TRIGger:SENT:STYPe? <SearchName>

Selects the serial message sequence to be searched for.

To enable the search for one of the serial message types, use `BUS<m>:SENT:SFORmat` and enable with `SEARch:TRIGger:SENT:SMSG`.

Parameters:

<SMFieldType> ID | IDDT

ID
Searches for the identifier.
To set the identifier condition for the serial message, use `SEARch:TRIGger:SENT:SICN`, `SEARch:TRIGger:SENT:SIMN` and `SEARch:TRIGger:SENT:SIMX`.

IDDT
Searches for the combination of identifier and data.
To set the identifier condition, use the commands shown above.
To set the data condition, use `SEARch:TRIGger:SENT:SDCN`, `SEARch:TRIGger:SENT:SDMN` and `SEARch:TRIGger:SENT:SDMX`.

*RST: ID

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:SIDType <SearchName>,<SMIDType>
SEARch:TRIGger:SENT:SIDType? <SearchName>

Sets the message ID format (4 bit or 8 bit) of the enhanced serial message.

Parameters:

<SMIDType> B4 | B8

*RST: B4

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:SICN <SearchName>,<SMIDOperator>
SEARch:TRIGger:SENT:SICN? <SearchName>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<SMIDOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one data pattern to be set with `SEARch:TRIGger:SENT:SIMN`.

INRange | OORange
In range / Out of range. To define the range set the minimum and maximum values with `SEARch:TRIGger:SENT:SIMN` and `SEARch:TRIGger:SENT:SIMX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:SIMN <SearchName>,<SMIDPattern>
SEARch:TRIGger:SENT:SIMN? <SearchName>

Specifies a message identifier pattern, or sets the start value of an identifier range.

Parameters:

<SMIDPattern> Numeric or string pattern, Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:SIMX <SearchName>,<SMIDPatternTo>
SEARch:TRIGger:SENT:SIMX? <SearchName>

Sets the end value of an identifier range for `SEARch:TRIGger:SENT:SICN INRange or OORange`.

Parameters:

<SMIDPatternTo> Numeric or string pattern, Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:SDCN <SearchName>,<SMDataOperator>
SEARch:TRIGger:SENT:SDCN? <SearchName>

Sets the operator to set a specific data pattern or a data pattern range.

Parameters:

<SMDataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one data pattern to be set with `SEARch:TRIGger:SENT:SDMN`.

INRange | OORange
In range / Out of range. To define the range set the minimum and maximum values with `SEARch:TRIGger:SENT:SDMN` and `SEARch:TRIGger:SENT:SDMX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:SDMN <SearchName>,<SMDataPattern>
SEARch:TRIGger:SENT:SDMN? <SearchName>

Specifies a data pattern, or sets the start value of a data pattern range.

Parameters:

<SMDataPattern> Numeric or string pattern, Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:SDMX <SearchName>,<SMDataPatternTo>
SEARch:TRIGger:SENT:SDMX? <SearchName>

Sets the end value of an identifier range for `SEARch:TRIGger:SENT:SDCN INRange or OORange`.

Parameters:

<SMDataPatternTo> Numeric or string pattern, Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:PULSeerror <SearchName>,<CalibPulseError>
SEARch:TRIGger:SENT:PULSeerror? <SearchName>

Enables the search for calibration pulse errors.

To initially enable the search for an error event, set `SEARch:TRIGger:SENT:ERRor` to ON.

Parameters:
<CalibPulseError> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:PPERioderror <SearchName>,<PulsePeriodErr>
SEARch:TRIGger:SENT:PPERioderror? <SearchName>

Enables the search for pulse period errors.

To initially enable the search for an error event, set `SEARch:TRIGger:SENT:ERRor` to ON.

Parameters:
<PulsePeriodErr> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:IRFLength <SearchName>,<IrregularFrmLen>
SEARch:TRIGger:SENT:IRFLength? <SearchName>

Enables the search for irregular frame length errors in transmission sequences.

This error type is relevant if pause pulse mode is set to constant frame length.

To initially enable the search for an error event, set `SEARch:TRIGger:SENT:ERRor` to ON.

Parameters:
<IrregularFrmLen> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:FORMerror <SearchName>,<FormError>
SEARch:TRIGger:SENT:FORMerror? <SearchName>

Enables the search for format errors in serial messages.

To initially enable the search for an error event, set `SEARch:TRIGger:SENT:ERRor` to ON.

Parameters:
<FormError> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SENT:CRCerror <SearchName>,<CRCError>
SEARch:TRIGger:SENT:CRCerror? <SearchName>

Enables the search for errors in the Cyclic Redundancy Check.

To initially enable the search for an error event, set `SEARch:TRIGger:SENT:ERRor` to ON.

Parameters:
<CRCError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

24.17.11.5 SENT search results

The search on decoded SENT data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 24.17.11.3, "Decode results", on page 1909.

SEARch:RESult:SENT:FCOunt?....1925
SEARch:RESult:SENT:FRAMe<m>:CSValue?....1925
SEARch:RESult:SENT:FRAMe<m>:DATA?....1925
SEARch:RESult:SENT:FRAMe<m>:IDTYpe?....1925
SEARch:RESult:SENT:FRAMe<m>:IDValue?....1926
SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:STATe?....1926
SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:VALue?....1926
SEARch:RESult:SENT:FRAMe<m>:PAPTicks?....1927
SEARch:RESult:SENT:FRAMe<m>:SCOM?....1927
SEARch:RESult:SENT:FRAMe<m>:SDATa?....1927
SEARch:RESult:SENT:FRAMe<m>:STARt?....1928
SEARch:RESult:SENT:FRAMe<m>:STATus?....1928
SEARch:RESult:SENT:FRAMe<m>:STOP?....1928
SEARch:RESult:SENT:FRAMe<m>:SYMBol?....1928
SEARch:RESult:SENT:FRAMe<m>:SYNCduration?....1929
SEARch:RESult:SENT:FRAMe<m>:TYPE?....1929

SEARch:RESult:SENT:FCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:

<Count>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:CSValue? <SearchName>

Returns the checksum value.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<ChecksumValue>	Range:	0 to 63
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:DATA? <SearchName>

Returns the data of the specified frame.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:IDTYpe? <SearchName>

Returns the identifier type of the selected frame.

Suffix:
<m> *
Selects the frame.

Query parameters:
<SearchName>

Return values:
<IdentifierType> B4 | B8

B4: standard format, 4 bit
B8: extended format, 8 bit

*RST: B4

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:IDValue? <SearchName>

Returns the identifier value of the selected frame.

Suffix:
<m> *

Selects the serial bus.

Query parameters:
<SearchName>

Return values:
<IdentifierValue> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:STATe? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<State> OK | UNDF

*RST: OK

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:VALue? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:

<Value>	Range:	0 to 15
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:PAPTicks? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<PausePulseTicks>	Range:	12 to 768
	Increment:	1
	*RST:	12

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:SCOM? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<StatusComm>	Range:	0 to 15
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:SDATa? <SearchName>

Returns the symbolic data of the specified frame.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SymbolicData> String that contain the symbolic data.

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:STATus? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameState> OK | SYNC | PULSe | CRC | IRFL | FORM | INSufficient
*RST: OK

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:STOP? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:SYMBol? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Translation>

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:SYNCduration? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SyncDuration> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:SENT:FRAMe<m>:TYPE? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameType> TRSQ | SMSG | EMSG
*RST: TRSQ

Usage: Query only

24.17.12 CXPI (option R&S RTO6-K520)

- Configuration....1929
- Trigger....1933
- Decode results....1941
- Search settings....1945
- Search result settings....1953

24.17.12.1 Configuration

BUS<m>:CXPI:BITRate:ENABle....1930
BUS<m>:CXPI:BITRate:VALue....1930
BUS<m>:CXPI:RESult:BITRate?....1930
BUS<m>:CXPI:DORD....1931
BUS<m>:CXPI:HYSTeresis....1931
BUS<m>:CXPI:POLarity....1931

BUS<m>:CXPI:THReshold....1931
BUS<m>:CXPI:SDATa....1932
BUS<m>:CXPI:IBS:MAX....1932
BUS<m>:CXPI:IBS:MIN....1932
BUS<m>:CXPI:IFS:MAX....1932
BUS<m>:CXPI:IFS:MIN....1933

BUS<m>:CXPI:BITRate:ENABle <BitrateEnable>

Enables the manual setting of a bit rate. You can set the bitrate value with `BUS<m>:CXPI:BITRate:VALue`.

Suffix:
<m> 1..4

Parameters:
<BitrateEnable> ON | OFF
*RST: OFF

BUS<m>:CXPI:BITRate:VALue <Bitrate>

Sets the number of transmitted bits per second. To set this value, you have to enable the bitrate first with `BUS<m>:CXPI:BITRate:ENABle`.

Suffix:
<m> 1..4

Parameters:
<Bitrate> Range: 1000 to 100000
Increment: 100
*RST: 19200
Default unit: bps

BUS<m>:CXPI:RESult:BITRate?

Queries the measured average bit rate, which is calculated if `BUS<m>:CXPI:BITRate:ENABle` is set to `OFF`.

Suffix:
<m> 1..4

Return values:
<AverageBitrate> Range: 1000 to 100000
Increment: 100
*RST: 19200
Default unit: bps

Usage: Query only

BUS<m>:CXPI:DORD <DisplayOrder>

Selects the order in which the signal is displayed in the honeycomb.

If `TRANsmitted` is selected, the signal is displayed in the order it occurs. If `LOGical` is selected, the signal is displayed according to the definition of the standard (MSB bit order).

Suffix:
<m> 1..4

Parameters:
<DisplayOrder> TRANsmitted | LOGical
*RST: TRANsmitted

BUS<m>:CXPI:HYSTeresis <ThresHyst>

Sets a value for the hysteresis.

Suffix:
<m> 1..4

Parameters:
<ThresHyst> Range: -12 to 12
Increment: 0.01
*RST: 1
Default unit: V

BUS<m>:CXPI:POLarity <Polarity>

Sets the polarity of the transmitted waveform to normal (high = 1) or inverted (low = 1).

Suffix:
<m> 1..4

Parameters:
<Polarity> NORMal | INVert
*RST: NORMal

BUS<m>:CXPI:THReshold <Threshold>

Sets the threshold value for the signal.

Suffix:
<m> 1..4

Parameters:
<Threshold> Range: -15 to 12
Increment: 0.01
*RST: 6
Default unit: V

BUS<m>:CXPI:SDATa <SourceData>

Selects the input channel for the signal.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Alternatively, digital channels can be used if MSO option R&S RTO6-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Suffix:
<m> 1..4

Parameters:
<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

For triggering on a serial bus, analog or digital channel sources are required.

*RST: C1W1

BUS<m>:CXPI:IBS:MAX <ExpectedIBSMax>

Sets the maximum for the inter-byte-space range.

Suffix:
<m> 1..4

Parameters:
<ExpectedIBSMax> Range: 0 to 15
Increment: 1
*RST: 10

BUS<m>:CXPI:IBS:MIN <ExpectedIBSMin>

Sets the minimum for the inter-byte-space range.

Suffix:
<m> 1..4

Parameters:
<ExpectedIBSMin> Range: 0 to 15
Increment: 1
*RST: 0

BUS<m>:CXPI:IFS:MAX <ExpectedIFSMax>

Sets the maximum for the inter-frame-space range.

Suffix:
<m> 1..4

Parameters:
<ExpectedIFSMax> Range: 16 to 100000
Increment: 1
*RST: 1000

BUS<m>:CXPI:IFS:MIN <ExpectedIFSMin>

Sets the minimum for the inter-frame-space range.

Suffix:
<m> 1..4

Parameters:
<ExpectedIFSMin> Range: 16 to 1000
Increment: 1
*RST: 21

24.17.12.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:CXPI:CT....1934
TRIGger<m>:CXPI:DATA:DCONdition....1934
TRIGger<m>:CXPI:DATA:DMAX....1934
TRIGger<m>:CXPI:DATA:DMIN....1935
TRIGger<m>:CXPI:DATA:ICONdition....1935
TRIGger<m>:CXPI:DATA:IMAX....1935
TRIGger<m>:CXPI:DATA:IMIN....1935
TRIGger<m>:CXPI:DEXTension:CONDition....1936
TRIGger<m>:CXPI:DEXTension:MAX....1936
TRIGger<m>:CXPI:DEXTension:MIN....1936
TRIGger<m>:CXPI:DLC:CONDition....1937
TRIGger<m>:CXPI:DLC:MAX....1937
TRIGger<m>:CXPI:DLC:MIN....1937
TRIGger<m>:CXPI:ERRor:CRC....1937
TRIGger<m>:CXPI:ERRor:DLC....1938
TRIGger<m>:CXPI:ERRor:IBS....1938
TRIGger<m>:CXPI:ERRor:IFS....1938
TRIGger<m>:CXPI:ERRor:LENGth....1938
TRIGger<m>:CXPI:ERRor:PARity....1939
TRIGger<m>:CXPI:ERRor:UART....1939

TRIGger<m>:CXPI:FID:CONDition 1939
TRIGger<m>:CXPI:FID:MAX 1940
TRIGger<m>:CXPI:FID:MIN 1940
TRIGger<m>:CXPI:NM 1940
TRIGger<m>:CXPI:TYPE 1940

TRIGger<m>:CXPI:CT <CT>

Sets the value of the counter (CT), 2 bits indicating the continuity of the frame.

Suffix:
<m> 1..3

Parameters:
<CT>

TRIGger<m>:CXPI:DATA:DCONdition <DataOperator>

Sets the condition for the data pattern. You can define no value, an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DataOperator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
Any data pattern is considered

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:CXPI:DATA:DMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:CXPI:DATA:DMIN` and `TRIGger<m>:CXPI:DATA:DMAX`.

*RST: OFF

TRIGger<m>:CXPI:DATA:DMAX <DataMax>

Sets the end value of a data type range if `TRIGger<m>:CXPI:DATA:DCONdition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<DataMax>

TRIGger<m>:CXPI:DATA:DMIN <DataMin>

Specifies a data bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DataMin>

TRIGger<m>:CXPI:DATA:ICONdition <DataIdxOperator>

Set the condition for the data index value. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `TRIGger<m>:CXPI:DATA:IMIN` command.

INRange = RANGe
In range: Set the minimum and maximum value using `TRIGger<m>:CXPI:DATA:IMIN` and `TRIGger<m>:CXPI:DATA:IMAX`.

*RST: INRange

TRIGger<m>:CXPI:DATA:IMAX <DataIndexMax>

Sets the end value of a data index range if `TRIGger<m>:CXPI:DATA:ICONdition` is set to `INRange` or `RANge`.

Suffix:
<m> 1..3

Parameters:
<DataIndexMax> Range: 1 to 0
Increment: 1
*RST: 0

TRIGger<m>:CXPI:DATA:IMIN <DataIndexMin>

Specifies a data index bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:

<DataIndexMin>	Range:	1 to 0
	Increment:	1
	*RST:	1

TRIGger<m>:CXPI:DEXTension:CONDition <DLCEXTOperator>

Set the condition for the DLC extend value. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DLCEXTOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:CXPI:DEXTension:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:CXPI:DEXTension:MIN` and `TRIGger<m>:CXPI:DEXTension:MAX`.

*RST: EQUal

TRIGger<m>:CXPI:DEXTension:MAX <DLCEXTMax>

Sets the end value of a DLC extend range if `TRIGger<m>:CXPI:DEXTension:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<DLCEXTMax>

TRIGger<m>:CXPI:DEXTension:MIN <DLCEXTMin>

Specifies a data extension bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DLCEXTMin>

TRIGger<m>:CXPI:DLC:CONDition <DLCOperator>

Set the condition for the DLC. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DLCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one DLC pattern to be set with the corresponding `TRIGger<m>:CXPI:DLC:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:CXPI:DLC:MIN` and `TRIGger<m>:CXPI:DLC:MAX`.

*RST: EQUal

TRIGger<m>:CXPI:DLC:MAX <DLCMax>

Sets the end value of a data type range if `TRIGger<m>:CXPI:DLC:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<DLCMax>

TRIGger<m>:CXPI:DLC:MIN <DLCMin>

Specifies a DLC pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DLCMin>

TRIGger<m>:CXPI:ERRor:CRC <ErrorCRC>

Triggers on a cyclic redundancy check error, if `TRIGger<m>:CXPI:TYPE` is set to `ERRor`.

The transmitting node calculates the cyclic redundancy check (CRC) value of a frame and stores it into the CRC field of the frame. CRC error occurs when this CRC field value differs from the value calculated by the receiving node.

Suffix:
<m> 1..3

Parameters:
<ErrorCRC> ON | OFF
*RST: ON

TRIGger<m>:CXPI:ERRor:DLC <ErrorDLC>

Triggers on a data length code (DLC) error, if `TRIGger<m>:CXPI:TYPE` is set to `ERRor`.

DLC error occurs when the value of the DLC field is different from the data byte field value.

Suffix:
<m> 1..3

Parameters:
<ErrorDLC> ON | OFF
*RST: ON

TRIGger<m>:CXPI:ERRor:IBS <ErrorIBS>

Triggers on an inter-byte-space (IBS) error, if `TRIGger<m>:CXPI:TYPE` is set to `ERRor`.

Suffix:
<m> 1..3

Parameters:
<ErrorIBS> ON | OFF
*RST: ON

TRIGger<m>:CXPI:ERRor:IFS <ErrorIFS>

Triggers on an inter-frame-space (IFS) error, if `TRIGger<m>:CXPI:TYPE` is set to `ERRor`.

Suffix:
<m> 1..3

Parameters:
<ErrorIFS> ON | OFF
*RST: ON

TRIGger<m>:CXPI:ERRor:LENGth <ErrorLength>

Triggers on a length error, if `TRIGger<m>:CXPI:TYPE` is set to `ERRor`.

Suffix:
<m> 1..3

Parameters:
<ErrorLength> ON | OFF
*RST: ON

TRIGger<m>:CXPI:ERRor:PARity <ErrorParity>

Triggers on a parity error indicating a transmission error, if `TRIGger<m>:CXPI:TYPE` is set to `ERRor`.

Suffix:
<m> 1..3

Parameters:
<ErrorParity> ON | OFF
*RST: ON

TRIGger<m>:CXPI:ERRor:UART <ErrorUart>

Triggers on an UART error, if `TRIGger<m>:CXPI:TYPE` is set to `ERRor`.

Suffix:
<m> 1..3

Parameters:
<ErrorUart> ON | OFF
*RST: ON

TRIGger<m>:CXPI:FID:CONDition <FrameIDOperator>

Set the condition for the frame ID. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<FrameIDOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:CXPI:FID:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:CXPI:FID:MIN` and `TRIGger<m>:CXPI:FID:MAX`.

*RST: EQUal

TRIGger<m>:CXPI:FID:MAX <FrameIDMax>

Sets the end value of a frame ID range if `TRIGger<m>:CXPI:FID:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<FrameIDMax>

TRIGger<m>:CXPI:FID:MIN <FrameIDMin>

Specifies a frame ID pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<FrameIDMin>

TRIGger<m>:CXPI:NM <NM>

Sets the value of the network management (NM) field.

Suffix:
<m> 1..3

Parameters:
<NM>

TRIGger<m>:CXPI:TYPE <Type>

Selects the trigger type for the CXPI analysis.

Suffix:
<m> 1..3

Parameters:
<Type> STARt | NORMal | NPOLI | SLEEP | LONG | LPOLI | PID | PTYPe | PTPD | ERRor

STARt
Triggers on the STARt of the frame.

NORMal
Triggers on a NORMal frame. Optionally you can define a frame ID condition, an NM, a CT, a DLC condition, a data pattern condition and an index condition.

NPOLI
Triggers on a Normal POLL frame. Optionally you can define a frame ID condition, an NM, a CT, a DLC condition, a data pattern condition and an index condition.

SLEEP
Triggers on a SLEEP frame. Optionally you can define an NM, a CT, a data pattern condition and an index condition.

LONG
Triggers on a Long frame. Optionally you can define a frame ID condition, an NM, a CT, a DLC extend condition, a data pattern condition and an index condition.

LPOLI
Triggers on a Long POLL frame.

PID
Triggers on a Protected ID field. Optionally, you can define the frame ID pattern.

PTYPe
Triggers on a Protected TYPE field.

PTPID
Triggers on a Protected Type field followed by a PID field. Optionally, you can define the frame ID pattern.

ERRor
Trigger on selected error types. Define the error types with:
`TRIGger<m>:CXPI:ERRor:CRC`
`TRIGger<m>:CXPI:ERRor:DLC`
`TRIGger<m>:CXPI:ERRor:IBS`
`TRIGger<m>:CXPI:ERRor:IFS`
`TRIGger<m>:CXPI:ERRor:LENGth`
`TRIGger<m>:CXPI:ERRor:PARity`
`TRIGger<m>:CXPI:ERRor:UART`

*RST: STARt

24.17.12.3 Decode results

To show the results on the screen, use the following commands:

- `BUS<m>:RESult` on page 1671

BUS<m>:CXPI:RESult:FCOunt?....1941
BUS<m>:CXPI:RESult:FRAMe<n>:DATA?....1942
BUS<m>:CXPI:RESult:FRAMe<n>:DLCV?....1942
BUS<m>:CXPI:RESult:FRAMe<n>:STARt?....1942
BUS<m>:CXPI:RESult:FRAMe<n>:STATe?....1943
BUS<m>:CXPI:RESult:FRAMe<n>:STOP?....1943
BUS<m>:CXPI:RESult:FRAMe<n>:TYPE?....1943
BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:STATus?....1944
BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:TYPE?....1944
BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:VALue?....1944

BUS<m>:CXPI:RESult:FCOunt?

Returns the number of frames in the acquisition.

Suffix:
<m> 1..4

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:DATA?

Returns the data bytes of the specified frame.

Suffix:
<m> 1..4

<n> *

Return values:
<Data>

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:DLCV?

Returns the data length code/ extension data length code field value.

Suffix:
<m> 1..4

<n> *

Return values:
<DLC> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:STARt?

Returns the start time of the specified frame.

Suffix:
<m> 1..4

<n> *

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:STATe?

Returns the state of the specified frame.

Suffix:
<m> 1..4

<n> *

Return values:
<State> OK | INComplete | ERR_IBS | ERR_IFS | ERR_LENGTH | ERR_CRC | ERR_PARITY | ERR_DLC | ERR_UART

*RST: OK

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:STOP?

Returns the stop time of the specified frame.

Suffix:
<m> 1..4

<n> *

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:TYPE?

Returns the type of the specified frame.

Suffix:
<m> 1..4

<n> *

Return values:
<Type> NORMal | NPOLI | SLEEP | LONG | LPOLI | PID | PTYPe | PTPID | UNKNown | IFS

NPOLI
Normal POLI

LPOLI
Long POLI

PID
Protected ID

PTPID
Protected Type + Protected ID

IFS
Inter-Frame-Space

*RST: NORMal

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:STATus?

Returns the status of the specified field.

Suffix:
<m> 1..4

<n> *

<o> *

Return values:
<Status> OK | INComplete | ERR_IBS | ERR_IFS | ERR_LENGTH | ERR_CRC | ERR_PARITY | ERR_DLC | ERR_UART

*RST: OK

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:TYPE?

Returns the type (name) of the specified field.

Suffix:
<m> 1..4

<n> *

<o> *

Return values:
<WordType>

Usage: Query only

BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:VALue?

Returns the numeric value of the specified field.

Suffix:
<m> 1..4

<n> *

<o> *

Return values:

<WordValue>	Range:	0 to 65535
	Increment:	1
	*RST:	0

Usage: Query only

24.17.12.4 Search settings

In search setup commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name. The commands are similar to CXPI trigger commands.

SEARch:TRIGger:CXPI:CT....1945
SEARch:TRIGger:CXPI:DATA:DCONdition....1946
SEARch:TRIGger:CXPI:DATA:DMAX....1946
SEARch:TRIGger:CXPI:DATA:DMIN....1946
SEARch:TRIGger:CXPI:DATA:ICONdition....1947
SEARch:TRIGger:CXPI:DATA:IMAX....1947
SEARch:TRIGger:CXPI:DATA:IMIN....1947
SEARch:TRIGger:CXPI:DEXTension:CONDition....1948
SEARch:TRIGger:CXPI:DEXTension:MAX....1948
SEARch:TRIGger:CXPI:DEXTension:MIN....1948
SEARch:TRIGger:CXPI:DLC:CONDition....1949
SEARch:TRIGger:CXPI:DLC:MAX....1949
SEARch:TRIGger:CXPI:DLC:MIN....1949
SEARch:TRIGger:CXPI:ERRor:CRC....1949
SEARch:TRIGger:CXPI:ERRor:DLC....1950
SEARch:TRIGger:CXPI:ERRor:IBS....1950
SEARch:TRIGger:CXPI:ERRor:IFS....1950
SEARch:TRIGger:CXPI:ERRor:LENGth....1950
SEARch:TRIGger:CXPI:ERRor:PARity....1951
SEARch:TRIGger:CXPI:ERRor:UART....1951
SEARch:TRIGger:CXPI:FID:CONDition....1951
SEARch:TRIGger:CXPI:FID:MAX....1952
SEARch:TRIGger:CXPI:FID:MIN....1952
SEARch:TRIGger:CXPI:NM....1952
SEARch:TRIGger:CXPI:TYPE....1952

SEARch:TRIGger:CXPI:CT <SearchName>,<CT>
SEARch:TRIGger:CXPI:CT? <SearchName>

Sets the value of the counter (CT), 2 bits indicating the continuity of the frame.

Parameters:
<CT>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DATA:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:CXPI:DATA:DCONdition? <SearchName>

Sets the condition for the data pattern. You can define no value, an exact value or a value range.

Parameters:

<DataOperator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
Any data pattern is considered

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:CXPI:DATA:DMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:CXPI:DATA:DMIN` and `SEARch:TRIGger:CXPI:DATA:DCONdition`.

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DATA:DMAX <SearchName>,<DataMax>
SEARch:TRIGger:CXPI:DATA:DMAX? <SearchName>

Sets the end value of a data type range if `SEARch:TRIGger:CXPI:DATA:DCONdition` is set to `INRange` or `OORange`.

Parameters:
<DataMax>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DATA:DMIN <SearchName>,<DataMin>
SEARch:TRIGger:CXPI:DATA:DMIN? <SearchName>

Specifies a data bit pattern, or sets the start value of a pattern range.

Parameters:
<DataMin>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DATA:ICONdition <SearchName>,<DataIdxOperator>
SEARch:TRIGger:CXPI:DATA:ICONdition? <SearchName>

Set the condition for the data index value. You can define an exact value or a value range.

Parameters:

<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `SEARch:TRIGger:CXPI:DATA:IMIN` command.

INRange = RANGe
In range: Set the minimum and maximum value using `SEARch:TRIGger:CXPI:DATA:IMIN` and `SEARch:TRIGger:CXPI:DATA:IMAX`.

*RST: INRange

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DATA:IMAX <SearchName>,<DataIndexMax>
SEARch:TRIGger:CXPI:DATA:IMAX? <SearchName>

Sets the end value of a data index range if `SEARch:TRIGger:CXPI:DATA:ICONdition` is set to `INRange` or `RANge`.

Parameters:

<DataIndexMax>
Range: 1 to 0
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DATA:IMIN <SearchName>,<DataIndexMin>
SEARch:TRIGger:CXPI:DATA:IMIN? <SearchName>

Specifies a data index bit pattern, or sets the start value of a pattern range.

Parameters:

<DataIndexMin>
Range: 1 to 0
Increment: 1
*RST: 1

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DEXTension:CONDition <SearchName>,<DLCEXTOperator>
SEARch:TRIGger:CXPI:DEXTension:CONDition? <SearchName>

Set the condition for the DLC extend value. You can define an exact value or a value range.

Parameters:

<DLCEXTOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:CXPI:DEXTension:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:CXPI:DEXTension:MIN` and `SEARch:TRIGger:CXPI:DEXTension:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DEXTension:MAX <SearchName>,<DLCEXTMax>
SEARch:TRIGger:CXPI:DEXTension:MAX? <SearchName>

Sets the end value of a DLC extend range if `SEARch:TRIGger:CXPI:DEXTension:CONDition` is set to `INRange` or `OORange`.

Parameters:
<DLCEXTMax>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DEXTension:MIN <SearchName>,<DLCEXTMin>
SEARch:TRIGger:CXPI:DEXTension:MIN? <SearchName>

Specifies a data extension bit pattern, or sets the start value of a pattern range.

Parameters:
<DLCEXTMin>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DLC:CONDition <SearchName>,<DLCOperator>
SEARch:TRIGger:CXPI:DLC:CONDition? <SearchName>

Set the condition for the DLC. You can define an exact value or a value range.

Parameters:

<DLCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one DLC pattern to be set with the corresponding `SEARch:TRIGger:CXPI:DLC:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:CXPI:DLC:MIN` and `SEARch:TRIGger:CXPI:DLC:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DLC:MAX <SearchName>,<DLCMax>
SEARch:TRIGger:CXPI:DLC:MAX? <SearchName>

Sets the end value of a data type range if `SEARch:TRIGger:CXPI:DLC:CONDition` is set to `INRange` or `OORange`.

Parameters:
<DLCMax>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:DLC:MIN <SearchName>,<DLCMin>
SEARch:TRIGger:CXPI:DLC:MIN? <SearchName>

Specifies a DLC pattern, or sets the start value of a pattern range.

Parameters:
<DLCMin>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:ERRor:CRC <SearchName>,<ErrorCRC>
SEARch:TRIGger:CXPI:ERRor:CRC? <SearchName>

Searches for a cyclic redundancy check error, if `SEARch:TRIGger:CXPI:TYPE` is set to `ERRor`.

Parameters:
<ErrorCRC> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:ERRor:DLC <SearchName>,<ErrorDLC>
SEARch:TRIGger:CXPI:ERRor:DLC? <SearchName>

Searches for a data length code error, if `SEARch:TRIGger:CXPI:TYPE` is set to `ERRor`.

Parameters:
<ErrorDLC> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:ERRor:IBS <SearchName>,<ErrorIBS>
SEARch:TRIGger:CXPI:ERRor:IBS? <SearchName>

Searches for an inter-byte-space error, if `SEARch:TRIGger:CXPI:TYPE` is set to `ERRor`.

Parameters:
<ErrorIBS> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:ERRor:IFS <SearchName>,<ErrorIFS>
SEARch:TRIGger:CXPI:ERRor:IFS? <SearchName>

Searches for an inter-frame-space error, if `SEARch:TRIGger:CXPI:TYPE` is set to `ERRor`.

Parameters:
<ErrorIFS> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:ERRor:LENGth <SearchName>,<ErrorLength>
SEARch:TRIGger:CXPI:ERRor:LENGth? <SearchName>

Searches for a length error, if `SEARch:TRIGger:CXPI:TYPE` is set to `ERRor`.

Parameters:
<ErrorLength> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:ERRor:PARity <SearchName>,<ErrorParity>
SEARch:TRIGger:CXPI:ERRor:PARity? <SearchName>

Searches for a parity error, if `SEARch:TRIGger:CXPI:TYPE` is set to `ERRor`.

Parameters:
<ErrorParity> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:ERRor:UART <SearchName>,<ErrorUart>
SEARch:TRIGger:CXPI:ERRor:UART? <SearchName>

Searches for an UART error, if `SEARch:TRIGger:CXPI:TYPE` is set to `ERRor`.

Parameters:
<ErrorUart> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:FID:CONDition <SearchName>,<FrameIDOperator>
SEARch:TRIGger:CXPI:FID:CONDition? <SearchName>

Set the condition for the frame ID. You can define an exact value or a value range.

Parameters:
<FrameIDOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:CXPI:FID:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:CXPI:FID:MIN` and `SEARch:TRIGger:CXPI:FID:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:FID:MAX <SearchName>,<FrameIDMax>
SEARch:TRIGger:CXPI:FID:MAX? <SearchName>

Sets the end value of a frame ID range if `SEARch:TRIGger:CXPI:FID:CONDition` is set to `INRange` or `OORange`.

Parameters:
<FrameIDMax>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:FID:MIN <SearchName>,<FrameIDMin>
SEARch:TRIGger:CXPI:FID:MIN? <SearchName>

Specifies a frame ID pattern, or sets the start value of a pattern range.

Parameters:
<FrameIDMin>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:NM <SearchName>,<NM>
SEARch:TRIGger:CXPI:NM? <SearchName>

Sets the value of the network management (NM) field.

Parameters:
<NM>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:CXPI:TYPE <SearchName>,<Type>
SEARch:TRIGger:CXPI:TYPE? <SearchName>

Searches for the selected CXPI frame type.

Parameters:

<Type> STARt | NORMal | NPOLI | SLEEP | LONG | LPOLI | PID | PTYPe | PTPD | ERRor

STARt
Triggers on the STARt of the frame.

NORMal
Triggers on a NORMal frame. Optionally you can define a frame ID condition, a NM, a CT, a DLC condition, a data pattern condition and an index condition.

NPOLI
Triggers on a Normal POLL frame. Optionally you can define a frame ID condition, a NM, a CT, a DLC condition, a data pattern condition and an index condition.

SLEEP
Triggers on a SLEEP frame. Optionally you can define a NM, a CT, a data pattern condition and an index condition.

LONG
Triggers on a Long frame. Optionally you can define a frame ID condition, a NM, a CT, a DLC extend condition, a data pattern condition and an index condition.

LPOLI
Triggers on a Long POLL frame.

PID
Triggers on a Protected ID field. Optionally, you can define the frame ID pattern.

PTYPe
Triggers on a Protected TYPE field.

PTPID
Triggers on a PType field followed by a PID field.Optionally, you can define the frame ID pattern.

ERRor
Trigger on selected error types. Define the error types with:
`SEARch:TRIGger:CXPI:ERRor:CRC` on page 1949
`SEARch:TRIGger:CXPI:ERRor:CRC` on page 1949
`SEARch:TRIGger:CXPI:ERRor:DLC` on page 1950
`SEARch:TRIGger:CXPI:ERRor:IBS` on page 1950
`SEARch:TRIGger:CXPI:ERRor:IFS` on page 1950
`SEARch:TRIGger:CXPI:ERRor:LENGth` on page 1950
`SEARch:TRIGger:CXPI:ERRor:PARity` on page 1951
`SEARch:TRIGger:CXPI:ERRor:UART` on page 1951

*RST: STARt

Parameters for setting and query:
<SearchName>

24.17.12.5 Search result settings

The search on decoded CXPI data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 24.17.12.3, "Decode results", on page 1941.

SEARch:RESult:CXPI:FCOunt?....1954
SEARch:RESult:CXPI:FRAMe<m>:DATA?....1954
SEARch:RESult:CXPI:FRAMe<m>:DLCV?....1954
SEARch:RESult:CXPI:FRAMe<m>:STARt?....1955
SEARch:RESult:CXPI:FRAMe<m>:STATe?....1955
SEARch:RESult:CXPI:FRAMe<m>:STOP?....1955
SEARch:RESult:CXPI:FRAMe<m>:TYPE?....1955
SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:STATus?....1956
SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:TYPE?....1956
SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:VALue?....1956

SEARch:RESult:CXPI:FCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:DATA? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:DLCV? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<DLC> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:STATe? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> OK | INComplete | ERR_IBS | ERR_IFS | ERR_LENGTH | ERR_CRC | ERR_PARITY | ERR_DLC | ERR_UART
*RST: OK

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:STOP? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:TYPE? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Type> NORMal | NPOLI | SLEEP | LONG | LPOLI | PID | PTYPe | PTPID | UNKNown | IFS

*RST: NORMal

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:STATus? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Status> OK | INComplete | ERR_IBS | ERR_IFS | ERR_LENGTH | ERR_CRC | ERR_PARITY | ERR_DLC | ERR_UART

*RST: OK

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:TYPE? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<WordType>

Usage: Query only

SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:VALue? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:

<WordValue> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

24.17.13 MIL-1553 (option R&S RTO6-K530)

- Configuration....1957
- Trigger....1960
- Decode results....1970
- Search settings....1973
- Search results....1980

24.17.13.1 Configuration

BUS<m>:MILStd:SOURce....1957
BUS<m>:MILStd:MAXResponse:BITS....1957
BUS<m>:MILStd:MAXResponse:SELect....1958
BUS<m>:MILStd:MINGap:SELect....1958
BUS<m>:MILStd:MINGap:BITS....1958
BUS<m>:MILStd:POLarity....1959
BUS<m>:MILStd:PRESet....1959
BUS<m>:MILStd:THReshold:HIGH....1959
BUS<m>:MILStd:THReshold:LOW....1959

BUS<m>:MILStd:SOURce <SourceData>

Sets the channel for the signal source.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: C1W1

BUS<m>:MILStd:MAXResponse:BITS <MaxResponseTime>

Sets the value for the maximum response time.

See also: `BUS<m>:MILStd:MAXResponse:SELect`.

Suffix:

<m> 1..4

Parameters:

<MaxResponseTime>	Range:	2E-6 to 262E-6
	Increment:	100E-9
	*RST:	14E-6
	Default unit:	s

BUS<m>:MILStd:MAXResponse:SELect <MaxRespSel>

Enables the detection of the maximum response time between the last bit of a word and the following status word sync during decoding.

To specify the maximum response time, use `BUS<m>:MILStd:MAXResponse:BITS` on page 1957.

Suffix:

<m>	1..4

Parameters:

<MaxRespSelect>	ON | OFF	
	*RST:	ON

BUS<m>:MILStd:MINGap:SELect <MinGapSelect>

Enables the detection of the minimum idle time between the last bit of a message and the following command word sync during decoding.

To specify the minimum gap, use `BUS<m>:MILStd:MINGap:BITS`.

Suffix:

<m>	1..4

Parameters:

<MinGapSelect>	ON | OFF	
	*RST:	ON

BUS<m>:MILStd:MINGap:BITS <MinGapTime>

Sets a value for the minimum gap.

See also: `BUS<m>:MILStd:MINGap:SELect`.

Suffix:

<m>	1..4

Parameters:

<MinGapTime>	Range:	2E-6 to 262E-6
	Increment:	100E-9
	*RST:	4E-6
	Default unit:	s

BUS<m>:MILStd:POLarity <Polarity>

Selects the wire on which the bus signal is measured.

Suffix:
<m> 1..4

Parameters:
<Polarity> NORMal | INVerted
*RST: NORMal

BUS<m>:MILStd:PRESet <Preset>

Sets the default threshold voltage.

Suffix:
<m> 1..4

Parameters:
<Preset> V05 | V2 | V5 | V7 | MANual
*RST: V5

BUS<m>:MILStd:THReshold:HIGH <ThresholdHigh>

Sets the lower threshold level of the signal.

Suffix:
<m> 1..4

Parameters:
<ThresholdHigh> Range: 0 to 14
Increment: 0.1
*RST: 5
Default unit: V

BUS<m>:MILStd:THReshold:LOW <ThresholdLow>

Sets the lower threshold level of the signal.

Suffix:
<m> 1..4

Parameters:
<ThresholdLow> Range: -14 to 0
Increment: 0.1
*RST: -5
Default unit: V

24.17.13.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:MILStd:TYPE....1961
TRIGger<m>:MILStd:DATA:RCONdition....1961
TRIGger<m>:MILStd:CMD:RCONdition....1961
TRIGger<m>:MILStd:CDST:RCONdition....1961
TRIGger<m>:MILStd:DATA:RMIN....1962
TRIGger<m>:MILStd:CMD:RMIN....1962
TRIGger<m>:MILStd:CDST:RMIN....1962
TRIGger<m>:MILStd:DATA:RMAX....1962
TRIGger<m>:MILStd:CMD:RMAX....1962
TRIGger<m>:MILStd:CDST:RMAX....1962
TRIGger<m>:MILStd:CMD:CCONdition....1962
TRIGger<m>:MILStd:CMD:CMAX....1963
TRIGger<m>:MILStd:CMD:CMIN....1963
TRIGger<m>:MILStd:CMD:SCONdition....1963
TRIGger<m>:MILStd:CMD:SMAX....1963
TRIGger<m>:MILStd:CMD:SMIN....1964
TRIGger<m>:MILStd:CMD:TR....1964
TRIGger<m>:MILStd:CDST:ICONdition....1964
TRIGger<m>:MILStd:CDST:IMAX....1965
TRIGger<m>:MILStd:CDST:IMIN....1965
TRIGger<m>:MILStd:DATA:DCONdition....1965
TRIGger<m>:MILStd:DATA:DMAX....1965
TRIGger<m>:MILStd:DATA:DMIN....1965
TRIGger<m>:MILStd:DATA:ICONdition....1966
TRIGger<m>:MILStd:DATA:IMAX....1966
TRIGger<m>:MILStd:DATA:IMIN....1966
TRIGger<m>:MILStd:ERRor:MANChester....1966
TRIGger<m>:MILStd:ERRor:PARity....1967
TRIGger<m>:MILStd:ERRor:SYNC....1967
TRIGger<m>:MILStd:MAXResponse:BITS....1967
TRIGger<m>:MILStd:MAXResponse:SELect....1967
TRIGger<m>:MILStd:MINGap:BITS....1968
TRIGger<m>:MILStd:MINGap:SELect....1968
TRIGger<m>:MILStd:STATus:BCReceived....1968
TRIGger<m>:MILStd:STATus:BUSY....1968
TRIGger<m>:MILStd:STATus:DBCaccept....1968
TRIGger<m>:MILStd:STATus:INSTrument....1969
TRIGger<m>:MILStd:STATus:MERRor....1969
TRIGger<m>:MILStd:STATus:SREQuest....1969

TRIGger<m>:MILStd:STATus:SUBSystem.....1969
TRIGger<m>:MILStd:STATus:TERMinal.....1969
TRIGger<m>:MILStd:TPSPecifier.....1969

TRIGger<m>:MILStd:TYPE <Type>

Sets the trigger type for MIL-1553 analysis.

Parameters:

<Type> STYPe | WTYPe | DATA | CDST | CMD | STATword | ERR

STYPe
SyncTYPe: triggers on a sync impulse.

WTYPe
WordTYPe: triggers on the selected word type.

DATA
Triggers on a data word that can be specified.

CDST
CommanDSTatus word: triggers on a command word or on a status word that can be specified.

CMD
CoMmanD word: triggers on a command word or on a status word that can be specified.

STATword
STATus word: triggers on a status word that can be specified.

ERR
ERRor: triggers on any combination of protocol errors.

*RST: STYPe

TRIGger<m>:MILStd:DATA:RCONdition <RTAOperator>
TRIGger<m>:MILStd:CMD:RCONdition <RTAOperator>
TRIGger<m>:MILStd:CDST:RCONdition <RTAOperator>

Sets the operator to define a remote terminal address:

- DATA: for data words
- CMD: for command words
- CDST: for status words

Parameters:

<RTAOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:MILStd:CDST:RMIN`.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MILStd:CDST:RMIN` and `TRIGger<m>:MILStd:CDST:RMAX`.

*RST: EQUal

TRIGger<m>:MILStd:DATA:RMIN <RTAPatternMin>
TRIGger<m>:MILStd:CMD:RMIN <RTAPatternMin>
TRIGger<m>:MILStd:CDST:RMIN <RTAPatternMin>

Specify a remote terminal address or set the the start value of a remote terminal address range:

- DATA: for data words
- CMD: for command words
- CDST: for status words

Parameters:

<RTAPatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:DATA:RMAX <RTAPatternMax>
TRIGger<m>:MILStd:CMD:RMAX <RTAPatternMax>
TRIGger<m>:MILStd:CDST:RMAX <RTAPatternMax>

Set the end value of a data range if `TRIGger<m>:MILStd:CDST:RCONdition` is set to `INRange` or `OORange`:

- DATA: for data words
- CMD: for command words
- CDST: for status words

Parameters:

<RTAPatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:CMD:CCONdition <WordCntOperator>

Sets the operator to set a specific data word count or mode code pattern.

Parameters:

<WordCntOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:MILStd:CMD:CMIN`.

INRange | OORange
In range/Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:MILStd:CMD:CMIN` and `TRIGger<m>:MILStd:CMD:CMAX`.

*RST: EQUal

TRIGger<m>:MILStd:CMD:CMAX <WordCntPattMax>

Sets the end value of a data word count/mode code pattern if `TRIGger<m>:MILStd:CMD:CCONdition` is set to `INRange` or `OORange`.

Parameters:

<WordCntPattMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:CMD:CMIN <WordCntPattMin>

Specifies a data word count/mode code pattern or sets the the start value of a pattern range.

Parameters:

<WordCntPattMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:CMD:SCONdition <SubaddrOperator>

Sets the operator to set a specific subaddress/mode pattern.

Parameters:

<SubaddrOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with`TRIGger<m>:MILStd:CMD:SMIN`.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MILStd:CMD:SMIN` and `TRIGger<m>:MILStd:CMD:SMAX`.

*RST: EQUal

TRIGger<m>:MILStd:CMD:SMAX <SubaddrPattMax>

Sets the end value of the subaddress range if `TRIGger<m>:MILStd:CMD:SCONdition` is set to `INRange` or `OORange`.

Parameters:

<SubaddrPattMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:CMD:SMIN <SubaddrPattMin>

Specifies a subaddress or sets the the start value of a subaddress range.

Parameters:

<SubaddrPattMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:CMD:TR <TRFlag>

Triggers on a transmission mode.

Parameters:

<TRFlag> ONE | ZERO | DC

ONE
Transmit direction.

ZERO
Receive direction.

DC
Either directions.

*RST: DC

TRIGger<m>:MILStd:CDST:ICONdition <InfoOperator>

Sets the operator to set a specific info for the 9th to 19th bit of a command or status word.

Parameters:

<InfoOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:MILStd:CDST:IMIN`.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MILStd:CDST:IMIN` and `TRIGger<m>:MILStd:CDST:IMAX`.

*RST: EQUal

TRIGger<m>:MILStd:CDST:IMAX <InfoPatternMax>

Sets the end value of the info range if `TRIGger<m>:MILStd:CDST:ICONdition` is set to `INRange`.

Parameters:

<InfoPatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:CDST:IMIN <InfoPatternMin>

Specifies an info or sets the the start value of an info range.

Parameters:

<InfoPatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:DATA:DCONdition <DataOperator>

Sets the operator to set a specific data pattern.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:MILStd:DATA:DMIN`.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MILStd:DATA:DMIN` and `TRIGger<m>:MILStd:DATA:DMAX`.

*RST: EQUal

TRIGger<m>:MILStd:DATA:DMAX <DataPatternMax>

Sets the end value of a data pattern range if `TRIGger<m>:MILStd:DATA:DCONdition` is set to `INRange` or `OORange`.

Parameters:

<DataPatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:DATA:DMIN <DataPatternMin>

Specifies a data pattern or sets the the start value of a data pattern range.

Parameters:

<DataPatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MILStd:DATA:ICONdition <DataIdxOperator>

Sets the operator to set a specific range within this series of the data words that is considered for the analysis.

Parameters:

<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal| LTHan | LETHan | GTHan | GETHan
Equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:MILStd:DATA:IMIN`.

INRange = RANGe
In range: set the minimum and maximum value of the range with `TRIGger<m>:MILStd:DATA:IMIN` and `TRIGger<m>:MILStd:DATA:IMAX`.

*RST: INRange

TRIGger<m>:MILStd:DATA:IMAX <DataIndexMax>

Sets the end value of a data word series index range if `TRIGger<m>:MILStd:DATA:ICONdition` is set to `INRange`.

Parameters:

<DataIndexMax> Range: 1 to 32
Increment: 1
*RST: 32

TRIGger<m>:MILStd:DATA:IMIN <DataIndexMin>

Specifies an index or sets the the start value of a data word series index range.

Parameters:

<DataIndexMin> Range: 1 to 32
Increment: 1
*RST: 1

TRIGger<m>:MILStd:ERRor:MANChester <ManCodingError>

Triggers on an error of the manchester coding if `TRIGger<m>:MILStd:TYPE` is set to `ERRor`.

Parameters:
<ManCodingError> ON | OFF
*RST: ON

TRIGger<m>:MILStd:ERRor:PARity <ParityError>

Triggers on even parity if `TRIGger<m>:MILStd:TYPE` is set to `ERRor`.

Parameters:
<ParityError> ON | OFF
*RST: ON

TRIGger<m>:MILStd:ERRor:SYNC <SyncError>

Triggers on an error of the synchronization if `TRIGger<m>:MILStd:TYPE` is set to `ERRor`.

Parameters:
<SyncError> ON | OFF
*RST: ON

TRIGger<m>:MILStd:MAXResponse:BITS <MaxResponseTime>

Sets the value for the maximum response time to be triggered on if `TRIGger<m>:MILStd:TYPE` is set to `ERRor`

Enable the error trigger on maximum response time with `TRIGger<m>:MILStd:MAXResponse:SELect`.

Parameters:
<MaxResponseTime> Range: 2E-6 to 262E-6
Increment: 100E-9
*RST: 14E-6
Default unit: s

TRIGger<m>:MILStd:MAXResponse:SELect <MaxRespSel>

Enables the trigger on exceeding the maximum response time if `TRIGger<m>:MILStd:TYPE` is set to `ERRor`. You can set the maximum time with: `TRIGger<m>:MILStd:MAXResponse:BITS`.

Parameters:
<MaxRespSel> ON | OFF
*RST: ON

TRIGger<m>:MILStd:MINGap:BITS <MinGapTime>

Sets the value for the minimum gap to be triggered on if `TRIGger<m>:MILStd:TYPE` is set to `ERRor`.

Parameters:

<MinGapTime>	Range:	2E-6 to 262E-6
	Increment:	100E-9
	*RST:	4E-6
	Default unit:	s

TRIGger<m>:MILStd:MINGap:SELect <MinGapSelect>

Enables triggering when the minimum gap is out of range if `TRIGger<m>:MILStd:TYPE` is set to `ERRor`. You can set the minimum gap with: `TRIGger<m>:MILStd:MINGap:BITS`.

Parameters:

<MinGapSelect>	ON | OFF	
	*RST:	ON

TRIGger<m>:MILStd:STATus:BCReceived <BroadcastCmd>

Triggers on the state of the broadcast command received bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<BroadcastCmd>	ONE | ZERO | DC	
	*RST:	DC

TRIGger<m>:MILStd:STATus:BUSY <BusyFlag>

Triggers on the state of the busy bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<BusyFlag>	ONE | ZERO | DC	
	*RST:	DC

TRIGger<m>:MILStd:STATus:DBCaccept <DynBusControl>

Triggers on the state of the dynamic bus control accept bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<DynBusControl>	ONE | ZERO | DC	
	*RST:	DC

TRIGger<m>:MILStd:STATus:INSTrument <InstFlag>

Triggers on the state of the instrumentation bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<InstFlag> ONE | ZERO | DC

*RST: ZERO

TRIGger<m>:MILStd:STATus:MERRor <MessageError>

Triggers on the state of the message error bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<MessageError> ONE | ZERO | DC

*RST: DC

TRIGger<m>:MILStd:STATus:SREQuest <ServiceRequest>

Triggers on the state of the the service request bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<ServiceRequest> ONE | ZERO | DC

*RST: DC

TRIGger<m>:MILStd:STATus:SUBSystem <SubsystemFlag>

Triggers on the state of the subsystem flag bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<SubsystemFlag> ONE | ZERO | DC

*RST: DC

TRIGger<m>:MILStd:STATus:TERMinal <TerminalFlag>

Triggers on the state of theterminal flag bit of the status word if `TRIGger<m>:MILStd:TYPE` is set to `STATword`.

Parameters:

<TerminalFlag> ONE | ZERO | DC

*RST: DC

TRIGger<m>:MILStd:TPSPecifier <TypeSpecifier>

Sets the sync impulse/ word type to be triggered on.

Parameters:

<TypeSpecifier> CSTatus | DATA | ALL

CSTatus: command/status word

*RST: ALL

24.17.13.3 Decode results

To load and acivate a label list, use:

- `BUS<m>:NEWList` on page 1672
- `BUS<m>:SYMBols` on page 1672

BUS<m>:MILStd:WCOunt?.....1970
BUS<m>:MILStd:WORD<n>:DATA?.....1970
BUS<m>:MILStd:WORD<n>:INFO?.....1971
BUS<m>:MILStd:WORD<n>:RTADdress?.....1971
BUS<m>:MILStd:WORD<n>:STARt?.....1971
BUS<m>:MILStd:WORD<n>:STATus?.....1972
BUS<m>:MILStd:WORD<n>:STOP?.....1972
BUS<m>:MILStd:WORD<n>:SYMBol?.....1972
BUS<m>:MILStd:WORD<n>:TYPE?.....1973

BUS<m>:MILStd:WCOunt?

Returns the number of decoded words.

Suffix:

<m> 1..4

Return values:

<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:MILStd:WORD<n>:DATA?

Return the data bytes of the specified word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<Data> 16-bit data of the specified word as a 2-byte bit pattern (B1, B2). The first byte B1 is the most significant byte.

Example:

```
BUS:MILStd:WORD4:DATA?
<-- #H08,#H49
```

Usage: Query only

BUS<m>:MILStd:WORD<n>:INFO?

Returns the info value for the specified word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<FrameInfo> Range: 0 to 2047
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:MILStd:WORD<n>:RTADdress?

Returns the RT address for the selected word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<FrameRta> Range: 0 to 31
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:MILStd:WORD<n>:STARt?

Return the start time of the selected word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<FrameStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:MILStd:WORD<n>:STATus?

Returns the overall state of the selected word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<FrameState> OK | SYNC | MANC | PAR | GAP | RT | INComplete | UNKNown

OK: the word is valid.
SYNC: synchronization error occured.
MANC: manchester coding error occured.
PAR: parity error occured.
GAP: timing gap error occured.
RT: remote terminal error occured.
INComplete: the sequence is not completely contained in the acquisition
UNKNown: unknown frame type

*RST: OK

Usage: Query only

BUS<m>:MILStd:WORD<n>:STOP?

Return the stop time of the selected word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<FrameStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:MILStd:WORD<n>:SYMBol?

Returns the label name of the word ID.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:
<Translation>

Usage: Query only

BUS<m>:MILStd:WORD<n>:TYPE?

Returns the type of the specified word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<FrameType> UNKNown | DATA | CMD | STAT | CMST | IM

CMD: command word
CMST: command/status word
IM: inter message. Shows if there are gap errors or response timeout.

*RST: DATA

Usage: Query only

24.17.13.4 Search settings

The search remote commands are very similar to the trigger commands. Therefore, search coommands are described in short, for more details, see the corresponding trigger command in Chapter 24.17.13.2, "Trigger", on page 1960.

SEARch:TRIGger:MILStd:TYPE 1974
SEARch:TRIGger:MILStd:DATA:RCONdition 1974
SEARch:TRIGger:MILStd:CMD:RCONdition 1974
SEARch:TRIGger:MILStd:CDST:RCONdition 1974
SEARch:TRIGger:MILStd:DATA:RMIN 1975
SEARch:TRIGger:MILStd:CMD:RMIN 1975
SEARch:TRIGger:MILStd:CDST:RMIN 1975
SEARch:TRIGger:MILStd:DATA:RMAX 1975
SEARch:TRIGger:MILStd:CMD:RMAX 1975
SEARch:TRIGger:MILStd:CDST:RMAX 1975
SEARch:TRIGger:MILStd:CDST:ICONdition 1976
SEARch:TRIGger:MILStd:CMD:CCONdition 1976
SEARch:TRIGger:MILStd:CMD:SCONdition 1976

SEARch:TRIGger:MILStd:DATA:DCONdition 1976
SEARch:TRIGger:MILStd:CDST:IMIN 1976
SEARch:TRIGger:MILStd:CMD:CMIN 1976
SEARch:TRIGger:MILStd:CMD:SMIN 1976
SEARch:TRIGger:MILStd:DATA:DMIN 1976
SEARch:TRIGger:MILStd:CDST:IMAX 1977
SEARch:TRIGger:MILStd:CMD:CMAX 1977
SEARch:TRIGger:MILStd:CMD:SMAX 1977
SEARch:TRIGger:MILStd:DATA:DMAX 1977
SEARch:TRIGger:MILStd:DATA:ICONdition 1977
SEARch:TRIGger:MILStd:DATA:IMIN 1977
SEARch:TRIGger:MILStd:DATA:IMAX 1978
SEARch:TRIGger:MILStd:CMD:TR 1978
SEARch:TRIGger:MILStd:ERRor:MANChester 1978
SEARch:TRIGger:MILStd:ERRor:PARity 1978
SEARch:TRIGger:MILStd:ERRor:SYNC 1978
SEARch:TRIGger:MILStd:ERRor:TIMing 1978
SEARch:TRIGger:MILStd:STATus:BCReceived 1979
SEARch:TRIGger:MILStd:STATus:BUSY 1979
SEARch:TRIGger:MILStd:STATus:DBCaccept 1979
SEARch:TRIGger:MILStd:STATus:INSTrument 1979
SEARch:TRIGger:MILStd:STATus:MERRor 1979
SEARch:TRIGger:MILStd:STATus:SREQuest 1979
SEARch:TRIGger:MILStd:STATus:SUBSystem 1979
SEARch:TRIGger:MILStd:STATus:TERMinal 1979
SEARch:TRIGger:MILStd:TPSPecifier 1979

SEARch:TRIGger:MILStd:TYPE <SearchName>,<Type>
SEARch:TRIGger:MILStd:TYPE? <SearchName>

Sets the event to be searched for.

Parameters:

<Type> STYPe | WTYPe | DATA | CDST | CMD | STATword | ERR

See `TRIGger<m>:MILStd:TYPE` on page 1961

*RST: STYPe

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MILStd:DATA:RCONdition <SearchName>,<RTAOperator>
SEARch:TRIGger:MILStd:DATA:RCONdition? <SearchName>
SEARch:TRIGger:MILStd:CMD:RCONdition <SearchName>,<RTAOperator>
SEARch:TRIGger:MILStd:CMD:RCONdition? <SearchName>
SEARch:TRIGger:MILStd:CDST:RCONdition <SearchName>,<RTAOperator>
SEARch:TRIGger:MILStd:CDST:RCONdition? <SearchName>

Set the operator to define a remote terminal address:

- DATA: for data words

- CMD: for command words
- CDST: for status words

Parameters:

<RTAOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with `SEARch:TRIGger:MILStd:CDST:RMIN`.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:MILStd:CDST:RMIN` and `SEARch:TRIGger:MILStd:CDST:RMAX` on page 1975.

*RST: EQUal

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:MILStd:DATA:RMIN <SearchName>,<RTAPatternMin>
SEARch:TRIGger:MILStd:DATA:RMIN? <SearchName>
SEARch:TRIGger:MILStd:CMD:RMIN <SearchName>,<RTAPatternMin>
SEARch:TRIGger:MILStd:CMD:RMIN? <SearchName>
SEARch:TRIGger:MILStd:CDST:RMIN <SearchName>,<RTAPatternMin>
SEARch:TRIGger:MILStd:CDST:RMIN? <SearchName>

Specify a remote terminal address or set the the start value of a remote terminal address range:

- DATA: for data words
- CMD: for command words
- CDST: for status words

Parameters:

<RTAPatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:MILStd:DATA:RMAX <SearchName>,<RTAPatternMax>
SEARch:TRIGger:MILStd:DATA:RMAX? <SearchName>
SEARch:TRIGger:MILStd:CMD:RMAX <SearchName>,<RTAPatternMax>
SEARch:TRIGger:MILStd:CMD:RMAX? <SearchName>
SEARch:TRIGger:MILStd:CDST:RMAX <SearchName>,<RTAPatternMax>
SEARch:TRIGger:MILStd:CDST:RMAX? <SearchName>

Set the end value of a data range if `SEARch:TRIGger:MILStd:CDST:RCONdition` is set to `INRange` or `OORange`:

- DATA: for data words
- CMD: for command words
- CDST: for status words

Parameters:

<RTAPatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:MILStd:CDST:ICONdition <SearchName>,<InfoOperator>
SEARch:TRIGger:MILStd:CDST:ICONdition? <SearchName>
SEARch:TRIGger:MILStd:CMD:CCONdition <SearchName>,<WordCntOperator>
SEARch:TRIGger:MILStd:CMD:CCONdition? <SearchName>
SEARch:TRIGger:MILStd:CMD:SCONdition <SearchName>,<SubaddrOperator>
SEARch:TRIGger:MILStd:CMD:SCONdition? <SearchName>
SEARch:TRIGger:MILStd:DATA:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:MILStd:DATA:DCONdition? <SearchName>

Sets the operator for the corresponding search:

- CDST:ICON - specific info for the 9th to 19th bit of a command or status word.
- CMD:CCON - specific data word count or mode code pattern in a command word
- CMD:SCON - specific subaddress/mode pattern in a command word
- DATA:DCON - data pattern in a data word

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

*RST: EQUal

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:MILStd:CDST:IMIN <SearchName>,<InfoPatternMin>
SEARch:TRIGger:MILStd:CDST:IMIN? <SearchName>
SEARch:TRIGger:MILStd:CMD:CMIN <SearchName>,<WordCntPattMin>
SEARch:TRIGger:MILStd:CMD:CMIN? <SearchName>
SEARch:TRIGger:MILStd:CMD:SMIN <SearchName>,<SubaddrPattMin>
SEARch:TRIGger:MILStd:CMD:SMIN? <SearchName>
SEARch:TRIGger:MILStd:DATA:DMIN <SearchName>,<DataPatternMin>
SEARch:TRIGger:MILStd:DATA:DMIN? <SearchName>

Sets the pattern or the start value of a pattern range for the corresponding search:

- CDST:IMIN - specific info for the 9th to 19th bit of a command or status word.
- CMD:CMIN - specific data word count or mode code pattern in a command word
- CMD:SMIN - specific subaddress/mode pattern in a command word
- DATA:DMIN - data pattern in a data word

Parameters:
<DataPatternMin>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MILStd:CDST:IMAX <SearchName>,<InfoPatternMax>
SEARch:TRIGger:MILStd:CDST:IMAX? <SearchName>
SEARch:TRIGger:MILStd:CMD:CMAX <SearchName>,<WordCntPattMax>
SEARch:TRIGger:MILStd:CMD:CMAX? <SearchName>
SEARch:TRIGger:MILStd:CMD:SMAX <SearchName>,<SubaddrPattMax>
SEARch:TRIGger:MILStd:CMD:SMAX? <SearchName>
SEARch:TRIGger:MILStd:DATA:DMAX <SearchName>,<DataPatternMax>
SEARch:TRIGger:MILStd:DATA:DMAX? <SearchName>

Sets the end value of a pattern range for the corresponding search:

- CDST:IMAX - specific info for the 9th to 19th bit of a command or status word.
- CMD:CMAX - specific data word count or mode code pattern in a command word
- CMD:SMAX - specific subaddress/mode pattern in a command word
- DATA:DMAX - data pattern in a data word

Parameters:
<DataPatternMax>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MILStd:DATA:ICONdition <SearchName>,<DataIdxOperator>
SEARch:TRIGger:MILStd:DATA:ICONdition? <SearchName>

Sets the operator to set a range within a series of the data words that is considered for the search.

Parameters:
<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

INRange = RANGe

*RST: INRange

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MILStd:DATA:IMIN <SearchName>,<DataIndexMin>
SEARch:TRIGger:MILStd:DATA:IMIN? <SearchName>

Specifies an index or sets the the start value of a data word series index range.

Parameters:

<DataIndexMin>	Range:	1 to 32
	Increment:	1
	*RST:	1

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MILStd:DATA:IMAX <SearchName>,<DataIndexMax>
SEARch:TRIGger:MILStd:DATA:IMAX? <SearchName>

Sets the end value of a data word series index range if the operator is set to `INRange`.

Parameters:

<DataIndexMax>	Range:	1 to 32
	Increment:	1
	*RST:	32

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MILStd:CMD:TR <SearchName>,<TRFlag>
SEARch:TRIGger:MILStd:CMD:TR? <SearchName>

Searches for a transmission mode.

Parameters:

<TRFlag>	ONE | ZERO | DC	
	*RST:	DC

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MILStd:ERRor:MANChester <SearchName>,<ManCodingError>
SEARch:TRIGger:MILStd:ERRor:MANChester? <SearchName>
SEARch:TRIGger:MILStd:ERRor:PARity <SearchName>,<ParityError>
SEARch:TRIGger:MILStd:ERRor:PARity? <SearchName>
SEARch:TRIGger:MILStd:ERRor:SYNC <SearchName>,<SyncError>
SEARch:TRIGger:MILStd:ERRor:SYNC? <SearchName>
SEARch:TRIGger:MILStd:ERRor:TIMing <SearchName>,<MinGapSelect>
SEARch:TRIGger:MILStd:ERRor:TIMing? <SearchName>

Enables search for errors if `SEARch:TRIGger:MILStd:TYPE` is set to `ERRor`.

- MANChester: error of the manchester coding
- PARity: even parity (parity error)
- SYNC: error of the synchronization
- TIMing: Minimum gap is out of range

Parameters:

<SyncError>, <ParityError>, <ManCodingError>, <MinGapSelect>

ON | OFF

*RST: ON

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:MILStd:STATus:BCReceived <SearchName>,<BroadcastCmd>
SEARch:TRIGger:MILStd:STATus:BCReceived? <SearchName>
SEARch:TRIGger:MILStd:STATus:BUSY <SearchName>,<BusyFlag>
SEARch:TRIGger:MILStd:STATus:BUSY? <SearchName>
SEARch:TRIGger:MILStd:STATus:DBCaccept <SearchName>,<DynBusControl>
SEARch:TRIGger:MILStd:STATus:DBCaccept? <SearchName>
SEARch:TRIGger:MILStd:STATus:INSTrument <SearchName>,<InstFlag>
SEARch:TRIGger:MILStd:STATus:INSTrument? <SearchName>
SEARch:TRIGger:MILStd:STATus:MERRor <SearchName>,<MessageError>
SEARch:TRIGger:MILStd:STATus:MERRor? <SearchName>
SEARch:TRIGger:MILStd:STATus:SREQuest <SearchName>,<ServiceRequest>
SEARch:TRIGger:MILStd:STATus:SREQuest? <SearchName>
SEARch:TRIGger:MILStd:STATus:SUBSystem <SearchName>,<SubsystemFlag>
SEARch:TRIGger:MILStd:STATus:SUBSystem? <SearchName>
SEARch:TRIGger:MILStd:STATus:TERMinal <SearchName>,<TerminalFlag>
SEARch:TRIGger:MILStd:STATus:TERMinal? <SearchName>

Specifies the values (0, 1, X) of the status flags if `SEARch:TRIGger:MILStd:TYPE` is set to `STATword`.

Parameters:

<TerminalFlag> ONE | ZERO | DC

*RST: DC

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:MILStd:TPSPecifier <SearchName>,<TypeSpecifier>
SEARch:TRIGger:MILStd:TPSPecifier? <SearchName>

Sets the sync impulse/ word type to be searched for.

Parameters:

<TypeSpecifier> CSTatus | DATA | ALL

CSTatus: command/status word

*RST: ALL

Parameters for setting and query:

<SearchName>

24.17.13.5 Search results

The search on decoded MIL-1553 data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 24.17.13.3, "Decode results", on page 1970.

SEARch:RESult:MILStd:WCOunt?....1980
SEARch:RESult:MILStd:WORD<m>:TYPE?....1980
SEARch:RESult:MILStd:WORD<m>:STATus?....1980
SEARch:RESult:MILStd:WORD<m>:STARt?....1981
SEARch:RESult:MILStd:WORD<m>:STOP?....1981
SEARch:RESult:MILStd:WORD<m>:SYMBol?....1981
SEARch:RESult:MILStd:WORD<m>:RTADdress?....1982
SEARch:RESult:MILStd:WORD<m>:DATA?....1982
SEARch:RESult:MILStd:WORD<m>:INFO?....1982

SEARch:RESult:MILStd:WCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:

<FrameCount>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:TYPE? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<FrameType>	UNKNown | DATA | CMD | STAT | CMST | IM	
	*RST:	DATA

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:STATus? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameState> OK | SYNC | MANC | PAR | GAP | RT | INComplete | UNKNown
*RST: OK

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:STOP? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:SYMBol? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Translation>

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:RTADdress? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameRta> Range: 0 to 31
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:DATA? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:MILStd:WORD<m>:INFO? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameInfo> Range: 0 to 2047
Increment: 1
*RST: 0

Usage: Query only

24.17.14 ARINC 429 (option R&S RTO6-K530)

- Configuration....1983
- Trigger....1986
- Decode results....1989
- Search settings....1993
- Search results....1996

24.17.14.1 Configuration

BUS<m>:ARINc:SOURce 1983
BUS<m>:ARINc:BRValue 1983
BUS<m>:ARINc:BRMode 1983
BUS<m>:ARINc:MAXGap:SELect 1984
BUS<m>:ARINc:MAXGap:BITS 1984
BUS<m>:ARINc:MINGap:SELect 1984
BUS<m>:ARINc:MINGap:BITS 1984
BUS<m>:ARINc:POLarity 1985
BUS<m>:ARINc:PRESet 1985
BUS<m>:ARINc:THReshold:HIGH 1985
BUS<m>:ARINc:THReshold:LOW 1985

BUS<m>:ARINc:SOURce <SourceData>

Sets the channel for the signal source.

Reference and math waveforms are only available if the trigger source is one of the input channels but not the serial bus.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4
*RST: C1W1

BUS<m>:ARINc:BRValue <BitRateValue>

Sets the number of transmitted bits per second.

Suffix:
<m> 1..4

Parameters:
<BitRateValue> Range: 10000 to 110000
Increment: 100
*RST: 100000
Default unit: bps

BUS<m>:ARINc:BRMode <BitRateMode>

Sets the bit rate mode to high or low speed.

Suffix:
<m> 1..4

Parameters:
<BitRateMode> HIGH | LOW
*RST: HIGH

BUS<m>:ARINc:MAXGap:SELect <MaxGapSelect>

Enables the detection of the maximum gap time during decoding.

To specify the minimum gap time `BUS<m>:ARINc:MINGap:BITS`.

Suffix:
<m> 1..4

Parameters:
<MaxGapSelect> ON | OFF
*RST: OFF

BUS<m>:ARINc:MAXGap:BITS <MaxGapBits>

Sets the value for the maximum gap between two words.

See also: `BUS<m>:ARINc:MAXGap:SELect`

Suffix:
<m> 1..4

Parameters:
<MaxGapBits> Range: 0 to 1000
Increment: 1
*RST: 100
Default unit: bit

BUS<m>:ARINc:MINGap:SELect <MinGapSelect>

Enables the detection of the minimum idle time between two words during decoding.

To specify the minimum gap, use`BUS<m>:ARINc:MINGap:BITS`.

Suffix:
<m> 1..4

Parameters:
<MinGapSelect> ON | OFF
*RST: ON

BUS<m>:ARINc:MINGap:BITS <MinGapBits>

Sets a value for the minimum timing gap between two words.

See also: `BUS<m>:ARINc:MINGap:SELect` on page 1984.

Suffix:
<m> 1..4

Parameters:

<MinGapBits>	Range:	0 to 100
	Increment:	1
	*RST:	4
	Default unit:	bit

BUS<m>:ARINc:POLarity <Polarity>

Sets the wire on which the bus signal is measured.

Suffix:

<m> 1..4

Parameters:

<Polarity> ALEG | BLEG

*RST: ALEG

BUS<m>:ARINc:PRESet <Preset>

Sets the default threshold voltage.

Suffix:

<m> 1..4

Parameters:

<Preset> V25 | V5 | MANual

*RST: V5

BUS<m>:ARINc:THReshold:HIGH <ThresholdHigh>

Sets the high threshold level of the signal.

Suffix:

<m> 1..4

Parameters:

<ThresholdHigh>	Range:	0 to 12
	Increment:	0.1
	*RST:	5
	Default unit:	V

BUS<m>:ARINc:THReshold:LOW <ThresholdLow>

Sets the low threshold level of the signal.

Suffix:

<m> 1..4

Parameters:

<ThresholdLow>	Range:	-12 to 0
	Increment:	0.1
	*RST:	-5
	Default unit:	V

24.17.14.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:ARINc:TYPE....1986
TRIGger<m>:ARINc:DATA:CONDition....1986
TRIGger<m>:ARINc:DATA:MIN....1987
TRIGger<m>:ARINc:DATA:MAX....1987
TRIGger<m>:ARINc:ERRor:CODing....1987
TRIGger<m>:ARINc:ERRor:PARity....1987
TRIGger<m>:ARINc:LABel:CONDition....1988
TRIGger<m>:ARINc:LABel:MIN....1988
TRIGger<m>:ARINc:LABel:MAX....1988
TRIGger<m>:ARINc:MINGap:SELect....1988
TRIGger<m>:ARINc:MINGap:BITS....1988
TRIGger<m>:ARINc:MAXGap:SELect....1989
TRIGger<m>:ARINc:MAXGap:BITS....1989
TRIGger<m>:ARINc:SDI....1989
TRIGger<m>:ARINc:SSM....1989

TRIGger<m>:ARINc:TYPE <Type>

Sets the trigger type for ARINC 429 analysis.

Parameters:

<Type>	STARt | STOP | LABel | ERRor	
	*RST:	STARt

TRIGger<m>:ARINc:DATA:CONDition <DataOperator>

Sets the condition for the data. You can define an exact data patterm or a data range.

Parameters:

<DataOperator>	EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:ARINc:DATA:MIN`.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:ARINc:DATA:MIN` and `TRIGger<m>:ARINc:DATA:MAX`.

*RST: EQUal

TRIGger<m>:ARINc:DATA:MIN <DataMin>

Defines the minimum bit pattern for the data.

Parameters:
<DataMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:ARINc:DATA:MAX <DataMax>

Sets the end value of a data pattern if `TRIGger<m>:ARINc:DATA:CONDition` is set to `INRange` or `OORange`.

Parameters:
<DataMax>

TRIGger<m>:ARINc:ERRor:CODing <CodingError>

Enables triggering when a coding error occurs.

Parameters:
<CodingError> ON | OFF

*RST: ON

TRIGger<m>:ARINc:ERRor:PARity <ParityError>

Enables triggering when a parity error occurs.

Parameters:
<ParityError> ON | OFF

*RST: ON

TRIGger<m>:ARINc:LABel:CONDition <LabelOperator>

Sets the condition for the label. You can define an exact label or a label range.

Parameters:

<LabelOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with`TRIGger<m>:ARINc:LABel:MIN`.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with`TRIGger<m>:ARINc:LABel:MIN` and `TRIGger<m>:ARINc:LABel:MAX`.

*RST: EQUal

TRIGger<m>:ARINc:LABel:MIN <LabelMin>

Defines the minimum bit pattern for the label.

Parameters:

<LabelMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:ARINc:LABel:MAX <LabelMax>

Sets the end value of a label pattern if `TRIGger<m>:ARINc:LABel:CONDition` is set to `INRange` or `OORange`.

Parameters:

<LabelMax> Numeric or string pattern, see `TRIGger<m>:ARINc:LABel:MIN`

TRIGger<m>:ARINc:MINGap:SELect <MinGapSelect>

Enables triggering when the minimum gap is out of range. You can set the minimum gap with: `TRIGger<m>:ARINc:MINGap:BITS`.

Parameters:

<MinGapSelect> ON | OFF

*RST: ON

TRIGger<m>:ARINc:MINGap:BITS <MinGapBits>

Sets the value for the minimum gap to be triggered on.

Parameters:

<MinGapBits>	Range:	0 to 100
	Increment:	1
	*RST:	4
	Default unit:	bit

TRIGger<m>:ARINc:MAXGap:SELect <MaxGapSelect>

Enables triggering when the maximum gap is out of range. You can set the maximum gap with: `TRIGger<m>:ARINc:MAXGap:BITS`.

Parameters:

<MaxGapSelect>	ON | OFF	
	*RST:	OFF

TRIGger<m>:ARINc:MAXGap:BITS <MaxGapBits>

Sets the value for the maximum gap to be triggered on.

Parameters:

<MaxGapBits>	Range:	0 to 1000
	Increment:	1
	*RST:	100
	Default unit:	bit

TRIGger<m>:ARINc:SDI <SDI>

Sets the source/destination identifier (SDI) bits.

Parameters:
<SDI>

TRIGger<m>:ARINc:SSM <SSM>

Sets the sign/status matrix (SSM) bits.

Parameters:
<SSM>

24.17.14.3 Decode results

To load and acivate a label list, use:

- `BUS<m>:NEWList` on page 1672
- `BUS<m>:SYMBols` on page 1672

BUS<m>:ARINc:WCOunt?.....1990
BUS<m>:ARINc:WORD<n>:DATA?.....1990
BUS<m>:ARINc:WORD<n>:LABel?.....1990
BUS<m>:ARINc:WORD<n>:PATTern?.....1991

BUS<m>:ARINc:WORD<n>:SDI?.....1991
BUS<m>:ARINc:WORD<n>:SSM?.....1991
BUS<m>:ARINc:WORD<n>:STARt?.....1992
BUS<m>:ARINc:WORD<n>:STATe?.....1992
BUS<m>:ARINc:WORD<n>:STOP?.....1992
BUS<m>:ARINc:WORD<n>:SYMBol?.....1993

BUS<m>:ARINc:WCOunt?

Returns the number of decoded words.

Suffix:
<m> 1..4

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:ARINc:WORD<n>:DATA?

Returns the data of the specified word.

Suffix:
<m> 1..4

<n> *

Return values:
<FrameData> 19-bit data field of the word as an integer

Range: 0 to 0
Increment: 1
*RST: 0

Example:
```
BUS:ARINc:WORD3:DATA?
<-- 148035
```

Usage: Query only

BUS<m>:ARINc:WORD<n>:LABel?

Returns the label of the specified word.

Suffix:
<m> 1..4

<n> *

Return values:
<FrameLabel> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:ARINc:WORD<n>:PATTern?

Returns all 32 bits of the specified word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *

Return values:

<Data> Comma-separated list of 4 bytes in big endian order. The format of each byte is defined by `FORMat:BPATtern`.

Example:

```
BUS2:ARINc:WORD3:PATTern?
--> #H75,#H11,#H55,#H82
FORMat:BPATtern DEC
BUS2:ARINc:WORD3:PATTern?
--> 117,17,85,130
```

Usage: Query only

BUS<m>:ARINc:WORD<n>:SDI?

Returns the source/destination identifier (SDI) bits of the specified word.

Suffix:

<m> 1..4

<n> *

Return values:

<SDI>
Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:ARINc:WORD<n>:SSM?

Returns the sign/status matrix(SSM) bits of the specified word.

Suffix:

<m> 1..4

<n> *

Return values:

<SSM>
Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:ARINc:WORD<n>:STARt?

Returns the start time of the specified word.

Suffix:

<m> 1..4

<n> *

Return values:

<FrameStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:ARINc:WORD<n>:STATe?

Returns the overall state of the specified word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the word.

Return values:

<FrameState> OK | CODE | GAP | PAR | INComplete | UNKNown

CODE: coding error occured.
GAP: timing gap error occured.
PAR: parity error occured.
INComplete: the sequence is not completely contained in the acquisition
UNKNown: unknown frame type

*RST: OK

Usage: Query only

BUS<m>:ARINc:WORD<n>:STOP?

Returns the end time of the specified word.

Suffix:

<m> 1..4

<n> *

Return values:

<FrameStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:ARINc:WORD<n>:SYMBol?

Returns the label name of the word ID.

Suffix:
<m> 1..4

<n> *

Return values:
<Translation>

Usage: Query only

24.17.14.4 Search settings

SEARch:TRIGger:ARINc:TYPE 1993
SEARch:TRIGger:ARINc:LABel:CONDition 1994
SEARch:TRIGger:ARINc:DATA:CONDition 1994
SEARch:TRIGger:ARINc:LABel:MIN 1994
SEARch:TRIGger:ARINc:DATA:MIN 1994
SEARch:TRIGger:ARINc:LABel:MAX 1994
SEARch:TRIGger:ARINc:DATA:MAX 1994
SEARch:TRIGger:ARINc:SDI 1995
SEARch:TRIGger:ARINc:SSM 1995
SEARch:TRIGger:ARINc:ERRor:CODing 1995
SEARch:TRIGger:ARINc:ERRor:PARity 1995
SEARch:TRIGger:ARINc:ERRor:TIMing 1996

SEARch:TRIGger:ARINc:TYPE <SearchName>,<Type>
SEARch:TRIGger:ARINc:TYPE? <SearchName>

Sets the search type.

Parameters:
<Type> STARt | STOP | LABel | ERRor
*RST: STARt

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:ARINc:LABel:CONDition <SearchName>,<LabelOperator>
SEARch:TRIGger:ARINc:LABel:CONDition? <SearchName>
SEARch:TRIGger:ARINc:DATA:CONDition <SearchName>,<DataOperator>
SEARch:TRIGger:ARINc:DATA:CONDition? <SearchName>

Set the condition for the label or data, respectively. You can define an exact value or a value range

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:ARINc:...:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range. with`TRIGger<m>:ARINc:LABel:MIN` and `TRIGger<m>:ARINc:LABel:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:ARINc:LABel:MIN <SearchName>,<LabelMin>
SEARch:TRIGger:ARINc:LABel:MIN? <SearchName>
SEARch:TRIGger:ARINc:DATA:MIN <SearchName>,<DataMin>
SEARch:TRIGger:ARINc:DATA:MIN? <SearchName>

Specifies a label or data bit pattern, or sets the the start value of a pattern range.

Parameters:

<DataMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ARINc:LABel:MAX <SearchName>,<LabelMax>
SEARch:TRIGger:ARINc:LABel:MAX? <SearchName>
SEARch:TRIGger:ARINc:DATA:MAX <SearchName>,<DataMax>
SEARch:TRIGger:ARINc:DATA:MAX? <SearchName>

Set the end value of a label or data pattern if the condition is set to `INRange` or `OORange`.

Parameters:

<DataMax> Numeric or string pattern, see `SEARch:TRIGger:ARINc:LABel:MIN`

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:ARINc:SDI <SearchName>,<SDI>
SEARch:TRIGger:ARINc:SDI? <SearchName>

Sets the source/destination identifier (SDI) bits.

Parameters:
<SDI>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:ARINc:SSM <SearchName>,<SSM>
SEARch:TRIGger:ARINc:SSM? <SearchName>

Sets the sign/status matrix (SSM) bits.

Parameters:
<SSM>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:ARINc:ERRor:CODing <SearchName>,<CodingError>
SEARch:TRIGger:ARINc:ERRor:CODing? <SearchName>

Enables the search for coding errors.

Parameters:
<CodingError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:ARINc:ERRor:PARity <SearchName>,<ParityError>
SEARch:TRIGger:ARINc:ERRor:PARity? <SearchName>

Enables the search for parity errors.

Parameters:
<ParityError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:ARINc:ERRor:TIMing <SearchName>,<MinGapSelect>
SEARch:TRIGger:ARINc:ERRor:TIMing? <SearchName>

Enables the search for timing errors, when the minimum gap is out of range.

Parameters:
<MinGapSelect> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

24.17.14.5 Search results

The search on decoded ARINC 429 data returns the same results as the queries for decode results.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 24.17.14.3, "Decode results", on page 1989.

SEARch:RESult:ARINc:WORD<m>:LABel?....1996
SEARch:RESult:ARINc:WORD<m>:PATTern?....1996
SEARch:RESult:ARINc:WORD<m>:DATA?....1997
SEARch:RESult:ARINc:WORD<m>:SSM?....1997
SEARch:RESult:ARINc:WORD<m>:SYMBol?....1997
SEARch:RESult:ARINc:WCOunt?....1998
SEARch:RESult:ARINc:WORD<m>:STOP?....1998
SEARch:RESult:ARINc:WORD<m>:SDI?....1998
SEARch:RESult:ARINc:WORD<m>:STATe?....1998
SEARch:RESult:ARINc:WORD<m>:STARt?....1999

SEARch:RESult:ARINc:WORD<m>:LABel? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Label> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:PATTern? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<DataValues>

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:DATA? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data> Range: 0 to 0
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:SSM? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SSM> Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:SYMBol? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Translation>

Usage: Query only

SEARch:RESult:ARINc:WCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:

<FrameCount>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:STOP? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<FrameStop>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:SDI? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<SDI>	Range:	0 to 3
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:STATe? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameState> OK | CODE | GAP | PAR | INComplete | UNKNown
*RST: OK

Usage: Query only

SEARch:RESult:ARINc:WORD<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

24.17.15 Space wire (option R&S RTO6-K530)

- Configuration....1999
- Trigger....2002
- Decode results....2005
- Search settings....2010
- Search results....2014

24.17.15.1 Configuration

BUS<m>:SWIRe:BPOSition....1999
BUS<m>:SWIRe:DATA:HYSTeresis....2000
BUS<m>:SWIRe:DATA:SOURce....2000
BUS<m>:SWIRe:DATA:THReshold....2000
BUS<m>:SWIRe:MGAP....2000
BUS<m>:SWIRe:STRBe:HYSTeresis....2001
BUS<m>:SWIRe:STRBe:SOURce....2001
BUS<m>:SWIRe:STRBe:THReshold....2001
BUS<m>:SWIRe:SYSLect....2002
BUS<m>:SWIRe:COUPling....2002
BUS<m>:SWIRe:PRESet....2002

BUS<m>:SWIRe:BPOSition <SyncBitPos>

Sets the bit position, the align position for the manual synchronisation mode. This can be useful when parity errors exist in the signal, and parity check is the main indicator for the decoder to do packet alignment.

Suffix:
<m> 1..4

Parameters:
<SyncBitPos> Range: 0 to 10
Increment: 1
*RST: 0

BUS<m>:SWIRe:DATA:HYSTeresis <HysteresisData>

Sets a value for the hysteresis of the data signal.

Suffix:
<m> 1..4

Parameters:
<HysteresisData> Range: 0 to 0.8
Increment: 0.01
*RST: 0.2
Default unit: V

BUS<m>:SWIRe:DATA:SOURce <SourceData>

Selects the source of the data signal.

Suffix:
<m> 1..4

Parameters:
<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15
*RST: C1W1

BUS<m>:SWIRe:DATA:THReshold <ThresholdData>

Sets the threshold value for the digitization of the data signal.

Suffix:
<m> 1..4

Parameters:
<ThresholdData> Range: -10 to 10
Increment: 0.1
*RST: 1.25
Default unit: V

BUS<m>:SWIRe:MGAP <MinGapTime>

Sets the minimum duration of a gap. Any inactivity greater than this time will be interpreted as a gap and lead to a resynchronization to the signal.

Suffix:
<m> 1..4

Parameters:
<MinGapTime> Range: 1E-9 to 10E-6
Increment: 100E-9
*RST: 200E-9
Default unit: s

Firmware/software: FW 3.30. Replaces `BUS<m>:SWIRe:MINGap`.

BUS<m>:SWIRe:STRBe:HYSTeresis <HystStrobe>

Sets a value for the hysteresis of the strobe signal.

Suffix:
<m> 1..4

Parameters:
<HystStrobe> Range: 0 to 0.8
Increment: 0.01
*RST: 0.2
Default unit: V

BUS<m>:SWIRe:STRBe:SOURce <SourceStrobe>

Selects the source of the strobe signal.

Suffix:
<m> 1..4

Parameters:
<SourceStrobe> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15
*RST: C2W1

BUS<m>:SWIRe:STRBe:THReshold <ThresholdStrobe>

Sets the threshold value for the digitization of the strobe signal.

Suffix:
<m> 1..4

Parameters:
<ThresholdStrobe> Range: -10 to 10
Increment: 0.1
*RST: 1.25
Default unit: V

BUS<m>:SWIRe:SYSLect <SyncSelector>

Selects the mode for the synchronisation settings, i.e i.e. packet align.

Suffix:
<m> 1..4

Parameters:
<SyncSelector> AUTO | MANual
*RST: AUTO

BUS<m>:SWIRe:COUPling <ThresCpl>

Enables coupling, i.e. the same threshold and hysteresis value is used for the strob and the data signal.

Suffix:
<m> 1..4

Parameters:
<ThresCpl> ON | OFF
*RST: ON

BUS<m>:SWIRe:PRESet <ThresholdPreset>

Prests the threshold and hysteresis value of the strobe and data signal.

Suffix:
<m> 1..4

Parameters:
<ThresholdPreset> V25 | V165 | V125 | V09 | V12 | V0 | MANual
V25: 2.5 V (CMOS 5.0 V)
V165: 1.65 V (CMOS 3.5 V)
V125: 1.25 V (CMOS 1.5 V)
V09: 2.5 V (CMOS 1.8V)
V12: 1.2 V (LVDS single ended)
V0: 0 V (LVDS differential)
MANual: user-defined value
*RST: V12

24.17.15.2 Trigger

TRIGger<m>:SWIRe:CTYPe....2003
TRIGger<m>:SWIRe:DATA:CONDition....2003
TRIGger<m>:SWIRe:DATA:MAX....2003
TRIGger<m>:SWIRe:DATA:MIN....2004
TRIGger<m>:SWIRe:ERRor:ESC....2004
TRIGger<m>:SWIRe:ERRor:PARity....2004
TRIGger<m>:SWIRe:TIME:CONDition....2004

TRIGger<m>:SWIRe:TIME:MAX....2005
TRIGger<m>:SWIRe:TIME:MIN....2005
TRIGger<m>:SWIRe:TYPE....2005

TRIGger<m>:SWIRe:CTYPe <ControlType>

Triggers on a specific control type character.

Suffix:
<m> 1..3

Parameters:
<ControlType> ANY | FCT | EOP | EEP

FCT
Flow Control Token

EOP
Normal End of Packet

EEP
Error End of Packet

*RST: ANY

TRIGger<m>:SWIRe:DATA:CONDition <DataOperator>

Set the condition for the data value. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:SWIRe:DATA:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:SWIRe:DATA:MIN` and `TRIGger<m>:SWIRe:DATA:MAX`.

*RST: EQUal

TRIGger<m>:SWIRe:DATA:MAX <DataPatternMax>

Sets the the end value of a data for the data pattern range if `TRIGger<m>:SWIRe:DATA:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<DataPatternMax>

TRIGger<m>:SWIRe:DATA:MIN <DataPatternMin>

Specifies a data for the data pattern, or sets the the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DataPatternMin>

TRIGger<m>:SWIRe:ERRor:ESC <ErrorESC>

Enables searching for an escape error.

Suffix:
<m> 1..3

Parameters:
<ErrorESC> ON | OFF
*RST: ON

TRIGger<m>:SWIRe:ERRor:PARity <ErrorParity>

Enables triggering on a parity error (even parity).

Suffix:
<m> 1..3

Parameters:
<ErrorParity> ON | OFF
*RST: ON

TRIGger<m>:SWIRe:TIME:CONDition <TimeOperator>

Set the condition for the data value for the time code. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<TimeOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:SWIRe:TIME:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:SWIRe:TIME:MIN` and `TRIGger<m>:SWIRe:TIME:MAX`.

*RST: EQUal

TRIGger<m>:SWIRe:TIME:MAX <TimePatternMax>

Sets the the end value of a data value for the time code range if `TRIGger<m>:SWIRe:TIME:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<TimePatternMax>

TRIGger<m>:SWIRe:TIME:MIN <TimePatternMin>

Specifies a pattern for the data value for the time code, or sets the the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<TimePatternMin>

TRIGger<m>:SWIRe:TYPE <Type>

Sets the trigger type for the SpaceWire analysis.

Suffix:
<m> 1..3

Parameters:
<Type> CTRL | DATA | NULL | TIME | ERRor

*RST: CTRL

24.17.15.3 Decode results

BUS<m>:SWIRe:RESults:FCOunt 2006
BUS<m>:SWIRe:RESults:FRAMe<n>:DATA? 2006
BUS<m>:SWIRe:RESults:FRAMe<n>:STARt? 2006
BUS<m>:SWIRe:RESults:FRAMe<n>:STATe? 2007
BUS<m>:SWIRe:RESults:FRAMe<n>:STOP? 2007
BUS<m>:SWIRe:RESults:FRAMe<n>:TYPE? 2007
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:FVAL? 2008
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STARt? 2008
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STATus? 2009

BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STOP?.....2009
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:TYPE?.....2010
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:VAL?.....2010

BUS<m>:SWIRe:RESults:FCOunt <Key>

Returns the number of decoded frames for the selected serial bus.

Suffix:
<m> 1..4

Setting parameters:
<Key>

Return values:
<Count>

BUS<m>:SWIRe:RESults:FRAMe<n>:DATA?

Returns the data value.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<Data> 8-bit data value

Example:
```
BUS:SWIRe:RESults:FRAMe2:DATA?
<-- #H12
```

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:STARt?

Returns the start time of the specified frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<Start>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:STATe?

Returns the overall state of the frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the serial bus.

Return values:

<State> OK | PAR | ESC | AMB | INComplete

OK
The frame is valid.

PARity
Parity error

ESC
Escape error

AMB
Ambiguous

INComplete
The frame is incomplete

*RST: OK

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:STOP?

Returns the end time of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:TYPE?

Returns the type of frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Type> DATA | FCT | EOP | EEP | ESC | NULL | TCOD

DATA
Data frame

FCT
Flow control token

EOP
End of packet

EEP
Error end of packet

ESC
Escape

NULL
Null symbol

TCOD
Time code

*RST: DATA

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:FVAL?

Returns the formatted value of the field with the selected field number within the frame with the selected frame number.

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<FrameByteValue>

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STARt?

Returns the start time of the field with the selected field number within the frame with the selected frame number.

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STATus?

Returns the status of the field with the selected field number within the frame with the selected frame number.

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<State> OK | PAR | ESC | AMB | INComplete

`PAR`: parity error
`ESC`: escape
`AMB`: ambiguous

*RST: OK

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STOP?

Returns the stop time of the field with the selected field number within the frame with the selected frame number.

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:TYPE?

Returns the type of the field with the selected field number within the frame with the selected frame number.

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<FrameByteValue>

Usage: Query only

BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:VAL?

Returns the value of the field with the selected field number within the frame with the selected frame number.

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<FrameByteValue> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

24.17.15.4 Search settings

SEARch:TRIGger:SWIRe:CTYPe 2011
SEARch:TRIGger:SWIRe:DATA:CONDition 2011
SEARch:TRIGger:SWIRe:DATA:MAX 2012
SEARch:TRIGger:SWIRe:DATA:MIN 2012

SEARch:TRIGger:SWIRe:ERRor:ESC 2012
SEARch:TRIGger:SWIRe:ERRor:PARity 2012
SEARch:TRIGger:SWIRe:TIME:CONDition 2013
SEARch:TRIGger:SWIRe:TIME:MAX 2013
SEARch:TRIGger:SWIRe:TIME:MIN 2013
SEARch:TRIGger:SWIRe:TYPE 2014

SEARch:TRIGger:SWIRe:CTYPe <SearchName>,<ControlType>
SEARch:TRIGger:SWIRe:CTYPe? <SearchName>

Searches for a specific control type character.

Parameters:

<ControlType> ANY | FCT | EOP | EEP

FCT
Flow Control Token

EOP
Normal End of Packet

EEP
Error End of Packet

*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SWIRe:DATA:CONDition <SearchName>,<DataOperator>
SEARch:TRIGger:SWIRe:DATA:CONDition? <SearchName>

Set the condition for the data pattern data value. You can define an exact value or a value range.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:SWIRe:DATA:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:SWIRe:DATA:MIN` and `SEARch:TRIGger:SWIRe:DATA:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SWIRe:DATA:MAX <SearchName>,<DataPatternMax>
SEARch:TRIGger:SWIRe:DATA:MAX? <SearchName>

Sets the the end value of a data type range if `SEARch:TRIGger:SWIRe:DATA:CONDition` is set to `INRange` or `OORange`.

Parameters:

<DataPatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SWIRe:DATA:MIN <SearchName>,<DataPatternMin>
SEARch:TRIGger:SWIRe:DATA:MIN? <SearchName>

Specifies a data bit pattern, or sets the the start value of a pattern range.

Parameters:
<DataPatternMin>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SWIRe:ERRor:ESC <SearchName>,<ErrorESC>
SEARch:TRIGger:SWIRe:ERRor:ESC? <SearchName>

Enables triggering on an escape error.

Parameters:

<ErrorESC> ON | OFF

*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SWIRe:ERRor:PARity <SearchName>,<ErrorParity>
SEARch:TRIGger:SWIRe:ERRor:PARity? <SearchName>

Enables searching for a parity error (even parity).

Parameters:

<ErrorParity> ON | OFF

*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SWIRe:TIME:CONDition <SearchName>,<TimeOperator>
SEARch:TRIGger:SWIRe:TIME:CONDition? <SearchName>

Set the condition for the data value of the time code. You can define an exact value or a value range.

Parameters:

<TimeOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:SWIRe:TIME:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:SWIRe:TIME:MIN` and `SEARch:TRIGger:SWIRe:TIME:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SWIRe:TIME:MAX <SearchName>,<TimePatternMax>
SEARch:TRIGger:SWIRe:TIME:MAX? <SearchName>

Sets the the end value of a data type range for the time code if `SEARch:TRIGger:SWIRe:TIME:CONDition` is set to `INRange` or `OORange`.

Parameters:
<TimePatternMax>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SWIRe:TIME:MIN <SearchName>,<TimePatternMin>
SEARch:TRIGger:SWIRe:TIME:MIN? <SearchName>

Specifies a data bit pattern for the time code, or sets the the start value of a pattern range.

Parameters:
<TimePatternMin>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:SWIRe:TYPE <SearchName>,<Type>
SEARch:TRIGger:SWIRe:TYPE? <SearchName>

Sets the search type for the SpaceWire analysis.

Parameters:
<Type> CTRL | DATA | NULL | TIME | ERRor
*RST: CTRL

Parameters for setting and query:
<SearchName>

24.17.15.5 Search results

In all `SEARch:RESult:DPHY:FRAMe<m>` commands, the suffix <m> selects the frame number in the list of search results.

SEARch:RESult:SWIRe:FCOunt?.....2014
SEARch:RESult:SWIRe:FRAMe<m>:DATA?.....2014
SEARch:RESult:SWIRe:FRAMe<m>:STARt?.....2015
SEARch:RESult:SWIRe:FRAMe<m>:STATe?.....2015
SEARch:RESult:SWIRe:FRAMe<m>:STOP?.....2015
SEARch:RESult:SWIRe:FRAMe<m>:TYPE?.....2016

SEARch:RESult:SWIRe:FCOunt? <SearchName>

Returns the number of frames within the search result for the selected serial bus.

Query parameters:
<SearchName>

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:SWIRe:FRAMe<m>:DATA? <SearchName>

Returns the data value for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:SWIRe:FRAMe<m>:STARt? <SearchName>

Returns the start time of the specified frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:SWIRe:FRAMe<m>:STATe? <SearchName>

Returns the overall state of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> OK | PAR | ESC | AMB | INComplete

OK
The frame is valid

PARity
Parity error

ESC
Escape Error

AMB
Ambiguous

INComplete
The frame is incomplete

*RST: OK

Usage: Query only

SEARch:RESult:SWIRe:FRAMe<m>:STOP? <SearchName>

Returns the end time for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Stop>

Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:SWIRe:FRAMe<m>:TYPE? <SearchName>

Returns the type of frame for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Type> DATA | FCT | EOP | EEP | ESC | NULL | TCOD

DATA
Data frame

FCT
Flow control token

EOP
End of packet

EEP
Error end of packet

ESC
Escape

NULL
Null symbol

TCOD
Time code

*RST: DATA

Usage: Query only

24.17.16 Ethernet 10BASE-T and 100BASE-TX (option R&S RTO6-K540)

- Configuration....2017
- Trigger....2019
- Decode results....2024
- Search settings....2030
- Search results....2035

24.17.16.1 Configuration

In all `BUS<m>:ETHernet` commands, the suffix <m> selects the serial bus.

BUS<m>:ETHernet:VARiant....2017
BUS<m>:ETHernet:SOURce....2017
BUS<m>:ETHernet:POLarity....2017
BUS<m>:ETHernet:THReshold:HIGH....2018
BUS<m>:ETHernet:THReshold:LOW....2018
BUS<m>:ETHernet:PRESet....2018
BUS<m>:ETHernet:BITRate....2019

BUS<m>:ETHernet:VARiant <Variant>

Selects the Ethernet protocol variant and transmission speed.

Suffix:
<m> 1..4

Parameters:
<Variant> B10T | B100TX | B100tx

B10T
Ethernet protocol variant 10BASE-T (10 Mbit/s)

B100TX = B100tx
Ethernet protocol variant 100BASE-TX (100 Mbit/s)

*RST: B10T

BUS<m>:ETHernet:SOURce <SourceData>

Selects the source channel for the data signal.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Alternatively, digital channels can be used if MSO option R&S RTO6-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Suffix:
<m> 1..4

Parameters:
<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

See Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C1W1

BUS<m>:ETHernet:POLarity <Polarity>

Defines the polarity of the data signal. This setting is only available in 10BASE-T.

Suffix:
<m> 1..4

Parameters:
<Polarity> NORMal | INVert

NORMal
Normal (non-inverted) data signal polarity

INVert
Inverted data signal polarity

*RST: NORMal

BUS<m>:ETHernet:THReshold:HIGH <ThresholdUpper>

Sets the upper threshold value for the signal digitization. If the signal value is higher than the this threshold, the signal state is considered high.

Suffix:
<m> 1..4

Parameters:
<ThresholdUpper> Range: 0 to 10
Increment: 0.01
*RST: 0.5
Default unit: V

BUS<m>:ETHernet:THReshold:LOW <ThresholdLower>

Sets the lower threshold value for the signal digitization. If the signal value is below this threshold, the signal state is considered low.

Suffix:
<m> 1..4

Parameters:
<ThresholdLower> Range: -10 to 0
Increment: 0.01
*RST: -0.5
Default unit: V

BUS<m>:ETHernet:PRESet <ThresholdPreset>

Sets the thresholds to predefined or individually definable voltage levels.

Suffix:
<m> 1..4

Parameters:
<ThresholdPreset> T0 | T100 | TX0 | TX100 | MANual

T0
Sets the thresholds to the default values for 10BASE-T (0 meters): upper threshold to 1.25 V, lower threshold to -1.25 V

T100
Sets the thresholds to the default values for 10BASE-T (100 meters): upper threshold to 0.75 V, lower threshold to -0.75 V

TX0
Sets the thresholds to the default values for 100BASE-TX (0 meters): upper threshold to 0.5 V, lower threshold to -0.5 V

TX100
Sets the thresholds to the default values for 100BASE-TX (100 meters): upper threshold to 0.35 V, lower threshold to -0.35 V

MANual
Allows to set individual threshold voltage levels

*RST: T0

BUS<m>:ETHernet:BITRate <BitRateValue>

Sets the bit rate value that defines the transmission speed in bits per second.

Suffix:
<m> 1..4

Parameters:
<BitRateValue> Range: 10000 to 150000000
Increment: 1000
*RST: 10000000
Default unit: bps

24.17.16.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:ETHernet:TYPE 2020
TRIGger<m>:ETHernet:PATTern 2020
TRIGger<m>:ETHernet:ERRor:CRC 2020
TRIGger<m>:ETHernet:ERRor:LENGth 2021
TRIGger<m>:ETHernet:ERRor:PREamble 2021
TRIGger<m>:ETHernet:FRAMe:CCONdition 2021
TRIGger<m>:ETHernet:FRAMe:CMAX 2022
TRIGger<m>:ETHernet:FRAMe:CMIN 2022
TRIGger<m>:ETHernet:FRAMe:DCONdition 2022
TRIGger<m>:ETHernet:FRAMe:DMAX 2022

TRIGger<m>:ETHernet:FRAMe:DMIN....2023
TRIGger<m>:ETHernet:FRAMe:SCONdition....2023
TRIGger<m>:ETHernet:FRAMe:SMAX....2023
TRIGger<m>:ETHernet:FRAMe:SMIN....2023
TRIGger<m>:ETHernet:FRAMe:TCONdition....2024
TRIGger<m>:ETHernet:FRAMe:TMAX....2024
TRIGger<m>:ETHernet:FRAMe:TMIN....2024

TRIGger<m>:ETHernet:TYPE <Type>

Selects the type of frame to be triggered on.

Suffix:
<m> 1..3

Parameters:
<Type> STARt | PATTern | ADVFrame | ADVCerror

STARt
Selects to trigger for the start of frame: search for the preamble and set the trigger instant thereafter.

PATTern
Selects to trigger for a bit pattern (data) to be specified in `TRIGger<m>:ETHernet:PATTern` on page 2020.

ADVFrame
Sets the trigger to the advanced frame. You can further define the values of the frame.

ADVCerror
Sets the trigger to advanced error. You can further enable the error conditions you want to trigger on.

*RST: STARt

TRIGger<m>:ETHernet:PATTern <DataPattern>

Specifies the bit pattern (or data) that is to be triggered.

Suffix:
<m> 1..3

Parameters:
<DataPattern>

TRIGger<m>:ETHernet:ERRor:CRC <ErrorCRC>

Enables the trigger on a mismatch of the cyclic redundancy check (CRC) value between the transmitting and receiving device.

Suffix:
<m> 1..3

Parameters:
<ErrorCRC> ON | OFF
*RST: ON

TRIGger<m>:ETHernet:ERRor:LENGth <ErrorLength>

Enables the trigger on an incorrect length of the sequence - when additional or missing bits are detected and the sequence of bits is not as expected.

Suffix:
<m> 1..3

Parameters:
<ErrorLength> ON | OFF
*RST: ON

TRIGger<m>:ETHernet:ERRor:PREamble <ErrorPreamble>

Enables the trigger on a frame with invalid preamble.

Suffix:
<m> 1..3

Parameters:
<ErrorPreamble> ON | OFF
*RST: ON

TRIGger<m>:ETHernet:FRAMe:CCONdition <CRCOperator>

Sets the operator to trigger on a specific frame check sequence value.

Suffix:
<m> 1..3

Parameters:
<CRCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a destination address to be set with `TRIGger<m>:ETHernet:FRAMe:CMIN`.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with `TRIGger<m>:ETHernet:FRAMe:CMIN` and `TRIGger<m>:ETHernet:FRAMe:CMAX`.

*RST: EQUal

TRIGger<m>:ETHernet:FRAMe:CMAX <CRCPatternMax>

Sets the end value of the frame check sequence range, if `TRIGger<m>:ETHernet:FRAMe:CCONdition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<CRCPatternMax>

TRIGger<m>:ETHernet:FRAMe:CMIN <CRCPatternMin>

Specifies a CRC pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<CRCPatternMin>

TRIGger<m>:ETHernet:FRAMe:DCONdition <DestAddrOptor>

Sets the destination address operator to trigger on a specific pattern or a range.

Suffix:
<m> 1..3

Parameters:
<DestAddrOptor> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a destination address to be set with `TRIGger<m>:ETHernet:FRAMe:DMIN`.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with `TRIGger<m>:ETHernet:FRAMe:DMIN` and `TRIGger<m>:ETHernet:FRAMe:DMAX`.

*RST: EQUal

TRIGger<m>:ETHernet:FRAMe:DMAX <DestAddrPattMax>

Sets the end value of a destination address range, if `TRIGger<m>:ETHernet:FRAMe:DMAX` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<DestAddrPattMax>

TRIGger<m>:ETHernet:FRAMe:DMIN <DestAddrPattMin>

Specifies a destination address pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DestAddrPattMin>

TRIGger<m>:ETHernet:FRAMe:SCONdition <SrcAddrOperator>

Sets source address operator to trigger on a specific pattern or a range.

Suffix:
<m> 1..3

Parameters:
<SrcAddrOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a destination address to be set with `TRIGger<m>:ETHernet:FRAMe:SMIN`.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with `TRIGger<m>:ETHernet:FRAMe:SMIN` and `TRIGger<m>:ETHernet:FRAMe:SMAX`.

*RST: EQUal

TRIGger<m>:ETHernet:FRAMe:SMAX <SrcAddrPattMax>

Sets the end value of a range, if the data condition is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<SrcAddrPattMax>

TRIGger<m>:ETHernet:FRAMe:SMIN <SrcAddrPattMin>

Specifies a source address pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<SrcAddrPattMin>

TRIGger<m>:ETHernet:FRAMe:TCONdition <TypeOperator>

Sets the type operator to trigger on a specific length/type value pattern or a range.

Suffix:
<m> 1..3

Parameters:
<TypeOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a destination address to be set with `TRIGger<m>:ETHernet:FRAMe:TMIN`.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with `TRIGger<m>:ETHernet:FRAMe:TMIN` and `TRIGger<m>:ETHernet:FRAMe:TMAX`.

*RST: EQUal

TRIGger<m>:ETHernet:FRAMe:TMAX <TypePatternMax>

Sets the end value of a length/type range, if `TRIGger<m>:ETHernet:FRAMe:TCONdition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<TypePatternMax>

TRIGger<m>:ETHernet:FRAMe:TMIN <TypePatternMin>

Specifies a length/type value, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<TypePatternMin>

24.17.16.3 Decode results

In all `BUS<m>:ETHernet:WORD<n>` commands, the suffix <m> selects the serial bus and the suffix <n> selects the word number in the decode table.

As an example, a set of query commands for bus #1 and word #1 is shown in the following, together with examples for results of these queries:

- `BUS1:ETH:WCOunt?` !2
- `BUS1:ETH:WORD1:STATe?` !OK

- `BUS1:ETH:WORD1:STARt?` !-0.000135
- `BUS1:ETH:WORD1:STOP?` !-6.62e-5
- `BUS1:ETH:WORD1:DEST?` !FF:FF:FF:FF:FF:FF
- `BUS1:ETH:WORD1:SRC?` !0F:0E:0D:0C:0B:0A
- `BUS1:ETH:WORD1:DATA?` ![60]45003c3e6210...
- `BUS1:ETH:WORD1:TYPE?` !2048
- `BUS1:ETH:WORD1:CRC?` !-1821935433
- `BUS1:ETH:WORD1:SSYM?` !
- `BUS1:ETH:WORD1:DSYM?` !BroadCast
- `BUS1:ETH:WORD1:BYTE1:VAL?` !69
- `BUS1:ETH:WORD1:BYTE2:VAL?` !0

BUS<m>:ETHernet:WCOunt?.....2025
BUS<m>:ETHernet:WORD<n>:FTYPe?.....2025
BUS<m>:ETHernet:WORD<n>:STATe?.....2026
BUS<m>:ETHernet:WORD<n>:STARt?.....2026
BUS<m>:ETHernet:WORD<n>:STOP?.....2026
BUS<m>:ETHernet:WORD<n>:DESTaddress?.....2027
BUS<m>:ETHernet:WORD<n>:SRCaddress?.....2027
BUS<m>:ETHernet:WORD<n>:TYPE?.....2027
BUS<m>:ETHernet:WORD<n>:DATA?.....2028
BUS<m>:ETHernet:WORD<n>:CRC?.....2028
BUS<m>:ETHernet:WORD<n>:DSYMbol?.....2028
BUS<m>:ETHernet:WORD<n>:SSYMbol?.....2029
BUS<m>:ETHernet:WORD<n>:BITRate?.....2029
BUS<m>:ETHernet:WORD<n>:BYTE<o>:VALue?.....2029
BUS<m>:ETHernet:WORD<n>:NUMWords?.....2030

BUS<m>:ETHernet:WCOunt?

Returns the frame count for the selected serial bus, i.e. the number of frames in the current acquisition. The result corresponds to the number of rows in the result table.

Suffix:
<m> 1..4

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:ETHernet:WORD<n>:FTYPe?

Returns the frame type of the selected word in the current acquisition.

Suffix:
<m> 1..4

<n> *

Return values:

<FrameType> MAC | IDLE | SLEep | EOS | UNKNown

*RST: MAC

Usage: Query only

BUS<m>:ETHernet:WORD<n>:STATe?

Returns the frame state of the selected word in the current acquisition.

Suffix:

<m> 1..4

<n> *

Return values:

<State> OK | ERR_PREAMBLE | ERR_LENGTH | UNCorrelated | INComplete

OK
No error detected

ERR_PREAMBLE
Error in the preamble of the selected word

ERR_SFD
Error in the start frame delimiter (SFD). The value of a correct SFD byte is 171. The SFD is transmitted LSB first.

ERR_LENGTH
Error in the number of bits in the selected word

*RST: OK

Usage: Query only

BUS<m>:ETHernet:WORD<n>:STARt?

Returns the frame start time of the selected word in the current acquisition.

Suffix:

<m> 1..4

<n> *

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
Default unit: s

Usage: Query only

BUS<m>:ETHernet:WORD<n>:STOP?

Returns the frame stop time of the selected word in the current acquisition.

Suffix:

<m>	1..4
<n>	*

Return values:

<Stop>	Range: -100E+24 to 100E+24 Increment: 100E-12 Default unit: s

Usage: Query only

BUS<m>:ETHernet:WORD<n>:DESTaddress?

Returns the destination address of the specified word.

Suffix:

<m>	1..4
<n>	*

Return values:

<DestAddress>	String parameter

Usage: Query only

BUS<m>:ETHernet:WORD<n>:SRCaddress?

Returns the source address of the specified word.

Suffix:

<m>	1..4
<n>	*

Return values:

<SrcAddress>	String parameter

Usage: Query only

BUS<m>:ETHernet:WORD<n>:TYPE?

The sub-protocol (e.g. HTML, video, etc.) determines what meaning this field has. Since the content of this data area is not decoded, the interpretation of the TYPE field is ambivalent.

Suffix:

<m>	1..4
<n>	*

Return values:

<Type> Returns the word type (specific for the sub-protocol), or the length of the selected word.

Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:ETHernet:WORD<n>:DATA?

Returns the number of word bytes in brackets [.] followed by the first six word bytes of data in hexadecimal format.

Use `BUS<m>:ETHernet:WORD<n>:BYTE<o>:VALue?` to access the word bytes.

Suffix:

<m> 1..4

<n> *

Return values:

<Data> String parameter

Example:

```
BUS:ETHernet:WORD3:DATA?
<-- '[60]FF00FFFF1234'
```

Usage: Query only

BUS<m>:ETHernet:WORD<n>:CRC?

Returns the Cyclic Redundancy Code (CRC, or frame check) checksum of the selected word.

Suffix:

<m> 1..4

<n> *

Return values:

<CRC> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:ETHernet:WORD<n>:DSYMbol?

Returns the symbolic label (or translation) of the destination address of the specified word, if the label list is enabled.

Suffix:

<m> 1..4

<n> *

Return values:
<DestTranslation> String parameter

Usage: Query only

BUS<m>:ETHernet:WORD<n>:SSYMbol?

Returns the symbolic label (or translation) of the source address of the specified word, if the label list is enabled.

Suffix:
<m> 1..4

<n> *

Return values:
<SrcTranslation> String parameter

Usage: Query only

BUS<m>:ETHernet:WORD<n>:BITRate?

Returns the primary bit rate.

Suffix:
<m> 1..4

<n> *

Return values:
<PrimaryBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

BUS<m>:ETHernet:WORD<n>:BYTE<o>:VALue?

BYTE returns all data of up to 1982 bytes (not just the first 5 or 6 bytes). This is also visible in the data table under "Show details".

Suffix:
<m> 1..4

<n> *

<o> *
Selects the byte number.

Return values:
<FrameByteValue> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:ETHernet:WORD<n>:NUMWords?

Returns the number of words in the selected frame. The result corresponds to the "Number of Words" column in the results table.

Suffix:
<m> 1..4

<n> *
Frame index

Return values:
<NumWords> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

24.17.16.4 Search settings

SEARch:TRIGger:ETHernet:TYPE....2030
SEARch:TRIGger:ETHernet:FRAMe:DCONdition....2031
SEARch:TRIGger:ETHernet:FRAMe:DMIN....2031
SEARch:TRIGger:ETHernet:FRAMe:DMAX....2031
SEARch:TRIGger:ETHernet:FRAMe:SCONdition....2032
SEARch:TRIGger:ETHernet:FRAMe:SMIN....2032
SEARch:TRIGger:ETHernet:FRAMe:SMAX....2032
SEARch:TRIGger:ETHernet:FRAMe:TCONdition....2033
SEARch:TRIGger:ETHernet:FRAMe:TMIN....2033
SEARch:TRIGger:ETHernet:FRAMe:TMAX....2033
SEARch:TRIGger:ETHernet:FRAMe:CCONdition....2034
SEARch:TRIGger:ETHernet:FRAMe:CMIN....2034
SEARch:TRIGger:ETHernet:FRAMe:CMAX....2034
SEARch:TRIGger:ETHernet:ERRor:PREamble....2035
SEARch:TRIGger:ETHernet:ERRor:LENGth....2035

SEARch:TRIGger:ETHernet:TYPE <SearchName>,<Type>
SEARch:TRIGger:ETHernet:TYPE? <SearchName>

Selects, if it searched on a MAC frame or on an error.

Parameters:
<Type> FRAMe | ERRor
*RST: FRAMe

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:ETHernet:FRAMe:DCONdition <SearchName>,<DestAddrOptor>
SEARch:TRIGger:ETHernet:FRAMe:DCONdition? <SearchName>

Defines the operator to search a specific destination address within a frame.

Parameters:

<DestAddrOptor> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a destination address to be set with `SEARch:TRIGger:ETHernet:FRAMe:DMIN`.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with `SEARch:TRIGger:ETHernet:FRAMe:DMIN` and `SEARch:TRIGger:ETHernet:FRAMe:DMAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:DMIN <SearchName>,<DestAddrPatt>
SEARch:TRIGger:ETHernet:FRAMe:DMIN? <SearchName>

Defines a destination address, or sets the start value of a destination address range.

Parameters:

<DestAddrPatt> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:DMAX <SearchName>,<DestAddrPatt>
SEARch:TRIGger:ETHernet:FRAMe:DMAX? <SearchName>

Sets the end value of a destination address range, if `SEARch:TRIGger:ETHernet:FRAMe:DCONdition` is set to `INRange` or `OORange`.

Parameters:

<DestAddrPatt> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:SCONdition <SearchName>,<SrcAddrOperator>
SEARch:TRIGger:ETHernet:FRAMe:SCONdition? <SearchName>

Defines the operator to search a specific source address within a frame.

Parameters:

<SrcAddrOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a destination address to be set with `SEARch:TRIGger:ETHernet:FRAMe:SMIN`.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with `SEARch:TRIGger:ETHernet:FRAMe:SMIN` and `SEARch:TRIGger:ETHernet:FRAMe:SMAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:SMIN <SearchName>,<SrcAddrPatt>
SEARch:TRIGger:ETHernet:FRAMe:SMIN? <SearchName>

Defines a source address, or sets the start value of a source address range.

Parameters:

<SrcAddrPatt> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:SMAX <SearchName>,<SrcAddrPatt>
SEARch:TRIGger:ETHernet:FRAMe:SMAX? <SearchName>

Sets the end value of a source address range, if `SEARch:TRIGger:ETHernet:FRAMe:SCONdition` is set to `INRange` or `OORange`.

Parameters:

<SrcAddrPatt> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:TCONdition <SearchName>,<TypeOperator>
SEARch:TRIGger:ETHernet:FRAMe:TCONdition? <SearchName>

Defines the operator to search for a specific frame length or type.

Parameters:

<TypeOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a pattern to be set with `SEARch:TRIGger:ETHernet:FRAMe:TMIN`.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with `SEARch:TRIGger:ETHernet:FRAMe:TMIN` and `SEARch:TRIGger:ETHernet:FRAMe:TMAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:TMIN <SearchName>,<TypePatternMin>
SEARch:TRIGger:ETHernet:FRAMe:TMIN? <SearchName>

Defines a frame length/type, or sets the start value for a range of frame lengths/types.

Parameters:

<TypePatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:TMAX <SearchName>,<TypePatternMax>
SEARch:TRIGger:ETHernet:FRAMe:TMAX? <SearchName>

Sets the end value of a range of frame lengths/types, if `SEARch:TRIGger:ETHernet:FRAMe:TCONdition` is set to `INRange` or `OORange`.

Parameters:

<TypePatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:CCONdition <SearchName>,<CRCOperator>
SEARch:TRIGger:ETHernet:FRAMe:CCONdition? <SearchName>

Defines the operator to search for a Cyclic Redundancy Code (CRC, or frame check) error condition within a frame.

Parameters:

<CRCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater Than, Greater than or equal. These conditions require a CRC pattern to be set with `SEARch:TRIGger:ETHernet:FRAMe:CMIN`.

INRange | OORange
In range, Out of range. Set the minimum and maximum value of the range with `SEARch:TRIGger:ETHernet:FRAMe:CMIN` and `SEARch:TRIGger:ETHernet:FRAMe:CMAX`.

*RST: EQUal

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:CMIN <SearchName>,<CRCPatternMin>
SEARch:TRIGger:ETHernet:FRAMe:CMIN? <SearchName>

Defines a CRC error condition pattern, or sets the start value of such a pattern.

Parameters:

<CRCPatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:FRAMe:CMAX <SearchName>,<CRCPatternMax>
SEARch:TRIGger:ETHernet:FRAMe:CMAX? <SearchName>

Sets the end value of a CRC error condition pattern, if `SEARch:TRIGger:ETHernet:FRAMe:CCONdition` is set to `INRange` or `OORange`.

Parameters:

<CRCPatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:ETHernet:ERRor:PREamble <SearchName>,<ErrorPreamble>
SEARch:TRIGger:ETHernet:ERRor:PREamble? <SearchName>

Defines, whether a search for any preamble error shall be activated or not.

Parameters:
<ErrorPreamble> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName> String parameter

SEARch:TRIGger:ETHernet:ERRor:LENGth <SearchName>,<ErrorLength>
SEARch:TRIGger:ETHernet:ERRor:LENGth? <SearchName>

Defines, whether a search for any word length error (too few or too many bits per word) shall be activated or not.

Parameters:
<ErrorLength> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName> String parameter

24.17.16.5 Search results

To show the results on the screen, use the following commands:

- `BUS<m>:RESult` on page 1671

In all `SEARch:RESult:ETHernet:WORD<m>` commands, the suffix <m> selects the word number in the list of search results.

SEARch:RESult:ETHernet:WCOunt?.....2035
SEARch:RESult:ETHernet:WORD<m>:STATe?.....2036
SEARch:RESult:ETHernet:WORD<m>:STARt?.....2036
SEARch:RESult:ETHernet:WORD<m>:STOP?.....2037
SEARch:RESult:ETHernet:WORD<m>:DESTaddress?.....2037
SEARch:RESult:ETHernet:WORD<m>:SRCaddress?.....2037
SEARch:RESult:ETHernet:WORD<m>:TYPE?.....2037
SEARch:RESult:ETHernet:WORD<m>:FTYPe?.....2038
SEARch:RESult:ETHernet:WORD<m>:DATA?.....2038
SEARch:RESult:ETHernet:WORD<m>:CRC?.....2038
SEARch:RESult:ETHernet:WORD<m>:DSYMbol?.....2039
SEARch:RESult:ETHernet:WORD<m>:SSYMbol?.....2039
SEARch:RESult:ETHernet:WORD<m>:BYTE<n>:VALue?.....2039

SEARch:RESult:ETHernet:WCOunt? <SearchName>

Returns the number of decoded words within the search result.

Query parameters:
<SearchName> String parameter

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:STATe? <SearchName>

Returns the frame state of the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName> String parameter

Return values:
<State> OK | ERR_PREAMBLE | ERR_LENGTH

OK
No error detected

ERR_PREAMBLE
Error in the preamble of the selected word

ERR_SFD
Error in the start frame delimiter (SFD). The value of a correct SFD byte is 171. The SFD is transmitted LSB first.

ERR_LENGTH
Error in the number of bits in the selected word

*RST: OK

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:STARt? <SearchName>

Returns the frame start time of the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName> String parameter

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:STOP? <SearchName>

Returns the frame stop time of the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName> String parameter

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:DESTaddress? <SearchName>

Returns the destination address of the specified word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName> String parameter

Return values:
<DestAddress> String parameter

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:SRCaddress? <SearchName>

Returns the source address of the specified word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName> String parameter

Return values:
<SourceAddress> String parameter

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:TYPE? <SearchName>

The sub-protocol (e.g. HTML, video, etc.) determines what meaning this field has. Since the content of this data area is not decoded, the interpretation of the TYPE field is ambivalent. The query either returns the word type (specific for the sub-protocol), or the length of the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName> String parameter

Return values:
<Type> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:FTYPe? <SearchName>

Returns the frame type of the specified frame.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameType> MAC | IDLE | SLEep | EOS | UNKNown
*RST: MAC

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:DATA? <SearchName>

Returns the data bytes of the specified word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName> String parameter

Return values:
<Data> String parameter

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:CRC? <SearchName>

Returns the Cyclic Redundancy Code (CRC, or frame check) checksum of the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName> String parameter

Return values:

<CRC>	Range:	0 to 4294967295
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:DSYMbol? <SearchName>

Returns the symbolic label (or translation) of the destination address of the specified word within the search result, if the label list is enabled.

Suffix:
<m> *

Query parameters:
<SearchName> String parameter

Return values:
<DestTranslation> String parameter

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:SSYMbol? <SearchName>

Returns the symbolic label (or translation) of the source address of the specified word within the search result, if the label list is enabled.

Suffix:
<m> *

Query parameters:
<SearchName> String parameter

Return values:
<SrcTranslation> String parameter

Usage: Query only

SEARch:RESult:ETHernet:WORD<m>:BYTE<n>:VALue? <SearchName>

BYTE returns all data of up to 1982 bytes (not just the first 5 or 6 bytes). This is also visible in the data table under "Show details".

Suffix:
<m> *

<n> *
Selects the byte number.

Query parameters:
<SearchName> String parameter

Return values:

<FrameByteValue> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

24.17.17 MDIO (option R&S RTO6-K540)

- Configuration....2040
- Trigger....2042
- Decode results....2045
- Search settings....2049
- Search results....2051

24.17.17.1 Configuration

In all `BUS<m>:MDIO` commands, the suffix <m> selects the serial bus.

BUS<m>:MDIO:CLOCk:SOURce....2040
BUS<m>:MDIO:DATA:SOURce....2041
BUS<m>:MDIO:CLOCk:THReshold:HIGH....2041
BUS<m>:MDIO:CLOCk:THReshold:LOW....2041
BUS<m>:MDIO:DATA:THReshold:HIGH....2041
BUS<m>:MDIO:DATA:THReshold:LOW....2042
BUS<m>:MDIO:PRESet....2042
BUS<m>:MDIO:COUPling....2042

BUS<m>:MDIO:CLOCk:SOURce <SourceClock>

Selects the source for the clock line (management data clock, MDC). Permitted selections are the analog channels "C1"–"C4" and the digital channels "D0"–"D15".

Suffix:

<m> 1..4

Parameters:

<SourceClock> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for data and clock lines. For triggering on a serial bus, analog or digital input channels are required.
See Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C2W1

BUS<m>:MDIO:DATA:SOURce <SourceData>

Selects the source for the data signal. Permitted selections are the analog channels "C1"–"C4" and the digital channels "D0"–"D15", but not the same as for "Clock".

Suffix:
<m> 1..4

Parameters:
<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Digital and analog channels cannot be used at the same time for data and clock lines. For triggering on a serial bus, analog or digital input channels are required.
See Chapter 24.4.2, "Waveform parameter", on page 1284

*RST: C1W1

BUS<m>:MDIO:CLOCk:THReshold:HIGH <ThresClkHigh>

Defines the upper threshold level for the clock signal.

Suffix:
<m> 1..4

Parameters:
<ThresClkHigh> Range: -5 to 5
Increment: 0.1
*RST: 2
Default unit: V

BUS<m>:MDIO:CLOCk:THReshold:LOW <ThresClkLow>

Defines the lower threshold level for the clock signal.

Suffix:
<m> 1..4

Parameters:
<ThresClkLow> Range: -5 to 5
Increment: 0.1
*RST: 0.8
Default unit: V

BUS<m>:MDIO:DATA:THReshold:HIGH <ThresDatHigh>

Defines the upper threshold level for the data signal.

Suffix:
<m> 1..4

Parameters:

<ThresDatHigh>	Range:	-5 to 5
	Increment:	0.1
	*RST:	2
	Default unit:	V

BUS<m>:MDIO:DATA:THReshold:LOW <ThresDatLow>

Defines the lower threshold level for the data signal.

Suffix:

<m>	1..4

Parameters:

<ThresDatLow>	Range:	-5 to 5
	Increment:	0.1
	*RST:	0.8
	Default unit:	V

BUS<m>:MDIO:PRESet <Preset>

Selects the default threshold settings according to the Ethernet standard: 2.0 V and 0.8 V.

Suffix:

<m>	1..4

Parameters:

<Preset>	DEFault | MANual	
	*RST:	DEFault

BUS<m>:MDIO:COUPling <ThresCpl>

Overwrites the data thresholds with the clock thresholds.

Suffix:

<m>	1..4

Parameters:

<ThresCpl>	ON | OFF	
	*RST:	ON

24.17.17.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.

- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:MDIO:TYPE....2043
TRIGger<m>:MDIO:ST....2043
TRIGger<m>:MDIO:FRAMetype....2044
TRIGger<m>:MDIO:PHYS....2044
TRIGger<m>:MDIO:REGI....2044
TRIGger<m>:MDIO:DATA....2045

TRIGger<m>:MDIO:TYPE <Type>

Selects the trigger type for MDIO analysis.

Parameters:

<Type> STARt | STOP | DATA

STARt
Sets the trigger to the start of frame (SOF) field. The start of frame condition and the trigger instant is the end of the preamble. Trigger pattern: preamble (32 bits "1")

STOP
Sets the trigger to the end of frame (EOF) field. The trigger instant is after the last data bit. Trigger pattern: preamble (32 bits "1") + 32 bits "X"

DATA
Sets the trigger to the data field. For more information on the data condition, see MDIO in the Ethernet standard. The trigger instant is at the end of the frame after the last data bit, as indicated in the GUI. Trigger pattern: preamble (32 bits "1") + 2 bits "ST" (Start of Frame Code) + 2 bits "OP" (Frame Type Code, or "OpCode") + 5 bits "PHYAD/PRTAD" (Physical Layer Entity Address / Port Address) + 5 bits "REGAD/DEVAD" (Register Address / Device Address) + 2 bits "TA" (turnaround time, X bits) + 16 bits "DATA/ADDRESS"

*RST: STARt

TRIGger<m>:MDIO:ST <StartCode>

Selects the start of frame code of the frame pattern; available only in trigger type "Data".

Note that Clause 22 is coded by "01", while Clause 45 is coded by "00", thus the lower Clause number is represented by the higher parameter value.

Parameters:

<StartCode> ST00 | ST01 | ST0X

ST00
Clause 45

ST01
Clause 22

ST0X
Any permissible start pattern

*RST: ST0X

TRIGger<m>:MDIO:FRAMetype <FrameType>

Selects the Type of Frame code (or OP code, OpCode, operation code); available only in trigger type "Data".

Note that the same OpCode may have different meanings in Clause 22 and Clause 45.

Parameters:
<FrameType> OP00 | OP01 | OP10 | OP11 | OPXX

OP00
Address frame (in Clause 45, only)

OP01
Write frame (in Clause 22 or Clause 45)

OP10
Read frame (in Clause 22) or Post-Read increment address frame (in Clause 45)

OP11
Read frame (in Clause 45)

OPXX
Any frame type

*RST: OPXX
Note that the user interface shows interpretations of the numerical OpCode values corresponding to Clause 45. Clause 22 is not represented by this interpretation.

TRIGger<m>:MDIO:PHYS <PhyAddr>

Sets the physical address (in Clause 22) or port address (in Clause 45) of the frame pattern (5 bits); available only in trigger type "Data".

Parameters:
<PhyAddr>

TRIGger<m>:MDIO:REGI <RegAddr>

Sets the register address (in Clause 22) or device address (in Clause 45) of the frame pattern (5 bits); available only in trigger type "Data".

Parameters:
<RegAddr>

TRIGger<m>:MDIO:DATA <Data>

Defines the 16-bit payload data pattern (both in Clause 22 or Clause 45) or the address pattern (in Clause 45, only) to trigger for; available only in trigger type "Data".

Parameters:
<Data>

24.17.17.3 Decode results

In all `BUS<m>:MDIO:WORD<n>` commands, the suffix <m> selects the serial bus and the suffix <n> selects the word number in the decode table.

BUS<m>:MDIO:WCOunt?....2045
BUS<m>:MDIO:WORD<n>:DATA?....2045
BUS<m>:MDIO:WORD<n>:PHYS?....2046
BUS<m>:MDIO:WORD<n>:REGI?....2046
BUS<m>:MDIO:WORD<n>:ST?....2046
BUS<m>:MDIO:WORD<n>:STARt?....2047
BUS<m>:MDIO:WORD<n>:STATe?....2047
BUS<m>:MDIO:WORD<n>:STOP?....2048
BUS<m>:MDIO:WORD<n>:SYMBol?....2048
BUS<m>:MDIO:WORD<n>:TYPE?....2048

BUS<m>:MDIO:WCOunt?

Returns the word count for the selected serial bus, i.e. the number of words in the present acquisition.

Suffix:
<m> 1..4

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:MDIO:WORD<n>:DATA?

Returns the 16-bit payload data field content (in Clause 22 or Clause 45), or the 16-bit address field content (in Clause 45, only) in the present acquisition of the selected word and the selected serial bus. The most significant bit (MSB) is transmitted first.

Suffix:
<m> 1..4

<n> *

Return values:
<Data>

Usage: Query only

BUS<m>:MDIO:WORD<n>:PHYS?

Returns the 5-bit address field content (PHYAD/PRTAD) in the present acquisition of the selected word and the selected serial bus.

Suffix:
<m> 1..4

<n> *

Return values:
<PhyAd> Range: 0 to 32
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:MDIO:WORD<n>:REGI?

Returns the 5-bit register or device address field content (REGAD/DEVAD) in the present acquisition of the selected word and the selected serial bus.

Suffix:
<m> 1..4

<n> *

Return values:
<RegAd> Range: 0 to 32
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:MDIO:WORD<n>:ST?

Returns the Start Code (= start of frame code) in the present acquisition of the selected word and the selected serial bus.

Suffix:
<m> 1..4

<n> *

Return values:
<StartCode> Range: 0 to 3
Increment: 1
*RST: 0
The parameter value "0" represents Clause 45, and "1" stands for Clause 22.
The values "2" and "3" do not correspond with any legal parameters according to the standard, but they can be searched for.

Usage: Query only

BUS<m>:MDIO:WORD<n>:STARt?

Returns the start time of the frame in the selected word of the selected serial bus.

Suffix:
<m> 1..4

<n> *

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:MDIO:WORD<n>:STATe?

Returns the state of the frame in the present acquisition of the selected serial bus.

Suffix:
<m> 1..4

<n> *

Return values:
<State> OK | UNSYN | UNSYncronized | OPCO | TA_ERROR | INComplete | SHORT | SHORt

OK
No error detected

UNSYN = UNSYncronized
UNSYncronized happens when bits are found, but they are not correlated with any synchronization sequence. We don't know what these bits are, but they are there. These bits receive a flag, but they are not decoded.

OPCO
OPcode Error

TA_ERROR
Tirnaround time error

INComplete
Incomplete Frame

SHORT = SHORt
Length Error

*RST: OK

Usage: Query only

BUS<m>:MDIO:WORD<n>:STOP?

Returns the stop time of the frame from the selected word within the search result.

Suffix:
<m> 1..4

<n> *

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:MDIO:WORD<n>:SYMBol?

Returns a textual translation (called Register Name) of the PHY or port address label in the present acquisition of the selected word and the selected serial bus.

The translation is defined in the label list.

Suffix:
<m> 1..4

<n> *

Return values:
<Translation>

Usage: Query only

BUS<m>:MDIO:WORD<n>:TYPE?

Returns the OpCode (= operation code or frame type) in the present acquisition of the selected word and the selected serial bus.

Suffix:
<m> 1..4

<n> *

Return values:
<FrameType> Range: 0 to 3
Increment: 1
*RST: 0
The parameter value "0" represents Clause 45, and "1" stands for Clause 22.
The values "2" and "3" do not correspond with any legal parameters according to the standard, but they can be searched for.

Usage: Query only

24.17.17.4 Search settings

SEARch:TRIGger:MDIO:DATA 2049
SEARch:TRIGger:MDIO:FRAMetype 2049
SEARch:TRIGger:MDIO:PHYS 2050
SEARch:TRIGger:MDIO:REGI 2050
SEARch:TRIGger:MDIO:ST 2050
SEARch:TRIGger:MDIO:TYPE 2051

SEARch:TRIGger:MDIO:DATA <SearchName>,<Data>
SEARch:TRIGger:MDIO:DATA? <SearchName>

Allows to define the 16-bit payload data pattern (both in Clause 22 or Clause 45) or the address pattern (in Clause 45, only) to search for; available only in search criteria type "Data".

Parameters:
<Data>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MDIO:FRAMetype <SearchName>,<FrameType>
SEARch:TRIGger:MDIO:FRAMetype? <SearchName>

Allows to select the Type of Frame code (or OP code, OpCode, operation code); available only in search criteria type "Data".

Note that the same OpCode may have different meanings in Clause 22 and Clause 45.

Parameters:
<FrameType> OP00 | OP01 | OP10 | OP11 | OPXX

OP00
Address frame (in Clause 45, only)

OP01
Write frame (in Clause 22 or Clause 45)

OP10
Read frame (in Clause 22) or Post-Read increment address frame (in Clause 45)

OP11
Read frame (in Clause 45)

OPXX
Any frame type

*RST: OPXX
Note that the user interface shows interpretations of the numerical OpCode values corresponding to Clause 45. Clause 22 is not represented by this interpretation.
Also, note that OPXX will never be a result of decoding, but it is still an option for triggering.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MDIO:PHYS <SearchName>,<PhyAddr>
SEARch:TRIGger:MDIO:PHYS? <SearchName>

Allows to set the physical address (in Clause 22) or port address (in Clause 45) of the frame pattern (5 bits); available only in search criteria type "Data".

Parameters:
<PhyAddr>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MDIO:REGI <SearchName>,<RegAddr>
SEARch:TRIGger:MDIO:REGI? <SearchName>

Allows to set the register address (in Clause 22) or device address (in Clause 45) of the frame pattern (5 bits); available only in search criteria type "Data".

Parameters:
<RegAddr>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MDIO:ST <SearchName>,<StartCode>
SEARch:TRIGger:MDIO:ST? <SearchName>

Allows to select the start of frame code of the frame pattern; available only in search criteria type "Data".

Parameters:
<StartCode> ST00 | ST01 | ST0X

ST00
Clause 45

ST01
Clause 22

ST0X
Any permissible start pattern

*RST: ST0X

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MDIO:TYPE <SearchName>,<Type>
SEARch:TRIGger:MDIO:TYPE? <SearchName>

Selects the event type to search for.

Parameters:
<Type> STARt | STOP | DATA

STARt
Searches for the start of frame (SOF).

STARt
Searches for the end of frame (EOF).

DATA
Allows to specify a payload data pattern (both in Clause 22 or Clause 45) or an address pattern (in Clause 45, only) to search for.

*RST: STARt

Parameters for setting and query:
<SearchName>

24.17.17.5 Search results

In all `SEARch:RESult:MDIO:WORD<m>` commands, the suffix <m> selects the packet number in the list of search results.

SEARch:RESult:MDIO:WORD<m>:DATA? 2051
SEARch:RESult:MDIO:WORD<m>:PHYS? 2052
SEARch:RESult:MDIO:WORD<m>:REGI? 2052
SEARch:RESult:MDIO:WORD<m>:ST? 2052
SEARch:RESult:MDIO:WORD<m>:STARt? 2053
SEARch:RESult:MDIO:WORD<m>:STATe? 2053
SEARch:RESult:MDIO:WORD<m>:STOP? 2054
SEARch:RESult:MDIO:WORD<m>:SYMBol? 2054
SEARch:RESult:MDIO:WORD<m>:TYPE? 2054
SEARch:RESult:MDIO:WCOunt? 2055

SEARch:RESult:MDIO:WORD<m>:DATA? <SearchName>

Returns the 16-bit payload data field content (in Clause 22 or Clause 45), or the 16-bit address field content (in Clause 45, only) from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:PHYS? <SearchName>

Returns the 5-bit address field content (PHYAD/PRTAD) from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<PhyAd> Range: 0 to 32
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:REGI? <SearchName>

Returns the 5-bit register or device address field content (REGAD/DEVAD) from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<RegAd> Range: 0 to 32
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:ST? <SearchName>

Returns the start of frame code from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<StartCode> Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:STARt? <SearchName>

Returns the start time of the frame from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:STATe? <SearchName>

Returns the state of the frame from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> OK | UNSYN | UNSYncronized | OPCO | TA_ERROR | INComplete | SHORT | SHORt

OK
No error detected

UNSYN = UNSYncronized
UNSYncronized happens when bits are found, but they are not correlated with any synchronization sequence. We don't know what these bits are, but they are there. These bits receive a flag, but they are not decoded.

OPCO
OPcode error

TA_ERROR
turnaround time error

INComplete
Incomplete Frame

SHORT = SHORt
Length Error

*RST: OK

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:STOP? <SearchName>

Returns the stop time of the frame from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Stop>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:SYMBol? <SearchName>

Returns a textual translation (called Register Name) of the PHY or port address label from the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Translation>

Usage: Query only

SEARch:RESult:MDIO:WORD<m>:TYPE? <SearchName>

Returns the frame type (= operation code or OpCode) for the selected word within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameType>
Range: 0 to 3
Increment: 1
*RST: 0
Note that the user interface shows interpretations of the numerical OpCode values corresponding to Clause 45: "0" (= [bin]00) represents Address, "1" (= [bin]01" represents Write, "2" (= [bin]10) represents Post Read, "3" (= [bin]11) represents Read. Clause 22 is not represented by this interpretation.

Usage: Query only

SEARch:RESult:MDIO:WCOunt? <SearchName>

Returns the word count within the search result.

Query parameters:
<SearchName>

Return values:

<FrameCount>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

24.17.18 RFFE (option R&S RTO6-K550)

For programming examples, see Chapter 24.3.7.1, "RFFE (option R&S RTO6-K550)", on page 1273.

24.17.18.1 Configuration

In all BUS<m>:RFFE commands, the suffix <m> selects the serial bus.

For programming examples, see "Configuring RFFE bus" on page 1273.

BUS<m>:RFFE:CLOCk:SOURce....2055
BUS<m>:RFFE:DATA:SOURce....2056
BUS<m>:RFFE:PRESet....2056
BUS<m>:RFFE:COUPling....2056
BUS<m>:RFFE:DATA:THReshold:HIGH....2056
BUS<m>:RFFE:DATA:THReshold:LOW....2057
BUS<m>:RFFE:DATA:THReshold:HYSTeresis....2057
BUS<m>:RFFE:CLOCk:THReshold:HIGH....2057
BUS<m>:RFFE:CLOCk:THReshold:LOW....2058
BUS<m>:RFFE:CLOCk:THReshold:HYSTeresis....2058
BUS<m>:RFFE:GFILter....2058
BUS<m>:RFFE:GFWidth....2059
BUS<m>:RFFE:MINGap:SELect....2059
BUS<m>:RFFE:MINGap:TIME....2059
BUS<m>:RFFE:RDMD....2059
BUS<m>:RFFE:VERSion....2060

BUS<m>:RFFE:CLOCk:SOURce <SourceClock>

Sets the source of the RFFE clock line.

For triggering on a serial bus, analog or digital input channels are required.

Suffix:
<m> 1..4

Parameters:
<SourceClock> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

*RST: C2W1

BUS<m>:RFFE:DATA:SOURce <SourceData>

Sets the source of the RFFE data line. For triggering on a serial bus, analog or digital input channels are required.

Suffix:
<m> 1..4

Parameters:
<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

*RST: C1W1

BUS<m>:RFFE:PRESet <ThresholdPreset>

Sets default threshold voltages for 1.2 V or 1.8 V bus. Set the value to MAN if you want to define individual thresholds.

Suffix:
<m> 1..4

Parameters:
<ThresholdPreset> V12 | V18 | MANual

*RST: V12

BUS<m>:RFFE:COUPling <ThresCpl>

If enabled, the SDATA thresholds are set to the SCLK threshold values.

Suffix:
<m> 1..4

Parameters:
<ThresCpl> ON | OFF

*RST: ON

BUS<m>:RFFE:DATA:THReshold:HIGH <ThresDatPos>

Set the positive going threshold (V_{TP}) for the data signal.

Note that the high and low thresholds and the hysteresis values are dependent, so it is sufficient to define two values for each line.

Suffix:
<m> 1..4

Parameters:
<ThresDatPos> Range: 0 to 2.5
Increment: 0.1
*RST: 0.72
Default unit: V

BUS<m>:RFFE:DATA:THReshold:LOW <ThresDatNeg>

Set the negative going threshold (V_{TN}) for the data signal.

Note that the high and low thresholds and the hysteresis values are dependent, so it is sufficient to define two values for each line.

Suffix:
<m> 1..4

Parameters:
<ThresDatNeg> Range: 0 to 2.5
Increment: 0.1
*RST: 0.48
Default unit: V

BUS<m>:RFFE:DATA:THReshold:HYSTeresis <ThresDatHyst>

Set the hysteresises between V_{TP} and V_{TN} for the data signal.

Note that the high and low thresholds and the hysteresis values are dependent, so it is sufficient to define two values for each line.

Suffix:
<m> 1..4

Parameters:
<ThresDatHyst> Range: 0 to 1
Increment: 0.1
*RST: 0.2
Default unit: V

BUS<m>:RFFE:CLOCk:THReshold:HIGH <ThresClkPos>

Set the positive going threshold (V_{TP}) for the clock signal.

Note that the high and low thresholds and the hysteresis values are dependent, so it is sufficient to define two values for each line.

Suffix:
<m> 1..4

Parameters:

<ThresClkPos>	Range:	0 to 2.5
	Increment:	0.1
	*RST:	0.72
	Default unit:	V

BUS<m>:RFFE:CLOCk:THReshold:LOW <ThresClkNeg>

Set the negative going threshold (V_{TN}) for the clock signal.

Note that the high and low thresholds and the hysteresis values are dependent, so it is sufficient to define two values for each line.

Suffix:

<m>	1..4

Parameters:

<ThresClkNeg>	Range:	0 to 2.5
	Increment:	0.1
	*RST:	0.48
	Default unit:	V

BUS<m>:RFFE:CLOCk:THReshold:HYSTeresis <ThresClkHyst>

Set the hysteresis between V_{TP} and V_{TN} for the clock signal.

Note that the high and low thresholds and the hysteresis values are dependent, so it is sufficient to define two values for each line.

Suffix:

<m>	1..4

Parameters:

<ThresClkHyst>	Range:	0 to 1
	Increment:	0.1
	*RST:	0.2
	Default unit:	V

BUS<m>:RFFE:GFILter <GlitchFilter>

Enables the glitch filter on the SCLK and SDATA lines to improve decode accuracy.

Suffix:

<m>	1..4 Selects the serial bus.

Parameters:

<GlitchFilter>	ON | OFF	
	*RST:	OFF

BUS<m>:RFFE:GFWidth <GlitchFilterWdt>

Sets the maximum glitch width to be ignored.

Suffix:
<m> 1..4

Parameters:
<GlitchFilterWdt> Range: 10E-12 to 10E-9
Increment: 10E-12
*RST: 1E-9
Default unit: s

BUS<m>:RFFE:MINGap:SELect <MinGapSelect>

If the minimum gap is enabled, the instrument detects the specified gap.

Set the minimum gap time using `BUS<m>:RFFE:MINGap:TIME`.

Suffix:
<m> 1..4

Parameters:
<MinGapSelect> ON | OFF
*RST: OFF

BUS<m>:RFFE:MINGap:TIME <MinGapTime>

Sets the minimum idle time between the Bus Park Cycle (BP) and Sequence Start Condition (SSC).

The setting is only relevant if `BUS<m>:RFFE:MINGap:SELect` is enabled.

Suffix:
<m> 1..4

Parameters:
<MinGapTime> Range: 1E-9 to 10E-6
Increment: 1E-9
*RST: 10E-9
Default unit: s

BUS<m>:RFFE:RDMD <ReadMode>

Selects, if the standard or synchronous read (`SREAD`) mode is used.

Suffix:
<m> 1..4

Parameters:
<ReadMode> STRD | SREAD
*RST: STRD

BUS<m>:RFFE:VERSion <Version>

Selects the RFFE version.

Suffix:
<m> 1..4

Parameters:
<Version> V1DX | V2D0 | V2D1 | V3D0

`V1DX`: 1.x
`V2D0`: 2.0
`V2D1`: 2.1
*RST: V1DX

24.17.18.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

For programming examples, see "Triggering on RFFE bus" on page 1274.

TRIGger<m>:RFFE:TYPE....2061
TRIGger<m>:RFFE:ADDRess:CONDition....2063
TRIGger<m>:RFFE:ADDRess:MIN....2063
TRIGger<m>:RFFE:ADDRess:MAX....2064
TRIGger<m>:RFFE:BCOunt:CONDition....2064
TRIGger<m>:RFFE:BCOunt:MIN....2064
TRIGger<m>:RFFE:BCOunt:MAX....2064
TRIGger<m>:RFFE:DATA:DCON....2065
TRIGger<m>:RFFE:DATA:DMIN....2065
TRIGger<m>:RFFE:DATA:DMAX....2065
TRIGger<m>:RFFE:DATA:ICONdition....2065
TRIGger<m>:RFFE:DATA:IMIN....2066
TRIGger<m>:RFFE:DATA:IMAX....2066
TRIGger<m>:RFFE:SADD:CONDition....2066
TRIGger<m>:RFFE:SADD:MIN....2067
TRIGger<m>:RFFE:SADD:MAX....2067
TRIGger<m>:RFFE:MASK:CONDition....2067
TRIGger<m>:RFFE:MASK:MAX....2067
TRIGger<m>:RFFE:MASK:MIN....2067
TRIGger<m>:RFFE:INTerrupt....2068
TRIGger<m>:RFFE:ERRor:BP....2068
TRIGger<m>:RFFE:ERRor:LENGth....2068
TRIGger<m>:RFFE:ERRor:NOResponse....2068

TRIGger<m>:RFFE:ERRor:PARity....2068
TRIGger<m>:RFFE:ERRor:SSC....2068
TRIGger<m>:RFFE:ERRor:USEQuence.... 2069
TRIGger<m>:RFFE:ERRor:GAP....2069
TRIGger<m>:RFFE:ERRor:VERSion.... 2069
TRIGger<m>:RFFE:INTerrupt....2069

TRIGger<m>:RFFE:TYPE <Type>

Selects the trigger type for RFFE analysis.

Parameters:

<Type> STARt | STOP | RZWR | RWR | RRD | ERWR | ERRD | ERWL | ERRL | IRSUM | MSKW | MASOHO | MASW | MASR | MCTW | MCTR | ERRor

STARt
Triggers on the beginning of a command sequence, exactly after the slave address. Optionally, you can specify a slave address condition.

STOP
Triggers on the end of a command sequence, exactly at the start of the bus park cycle. Optionally, you can specify a slave address condition.

RZWR
Triggers on Register 0 Write command sequences. Optionally, you can specify a slave address condition and a data pattern condition.

RWR
Triggers on Register Write command sequences. Optionally, you can specify a slave address condition, a register address condition, and a data pattern condition.

RRD
Triggers on Register Read command sequences. Optionally, you can specify a slave address condition, a register address condition, and a data pattern condition.

ERWR
Triggers on Extended Register Write command sequences. Optionally, you can specify a slave address condition, a register address condition, a byte count condition, a data index condition, and a data pattern condition.

ERRD
Triggers on Extended Register Read command sequences. Optional trigger conditions are the same as for ERWR.

ERWL
Triggers on Extended Register Write Long command sequences. Optional trigger conditions are the same as for ERWR.

ERRL
Triggers on Extended Register Read Long command sequences. Optional trigger conditions are the same as for ERWR.

IRSUM
Triggers on Interrupt Summary and Notification command sequences. Optionally, you can specify a slave address condition, bit count condition and a data pattern condition.

MSKW
Masked Write command sequence. Optionally, you can specify a slave address condition, register address condition, mask condition and data pattern condition

MASOHO
Triggers onMaster Ownership Handover command sequences. Optionally, you can specify a MID condition.

MASW
Triggers on Master Write command sequences. Optionally, you can specify a MID condition, a register address condition, and a data pattern condition.

MASR
Triggers on Master Read command sequences. Optionally, you can specify a MID condition, a register address condition, and a data pattern condition.

MCTW
Master Context Transfer Write command sequence. Optional: MID condition, register address condition, byte count condition, data index condition, and data pattern condition.

MCTR
Master Context Transfer Read command sequence. Optional search conditions are the same as for `MCTW`.

ERRor
Triggers if one of the enabled errors occurs in a frame.
To enable the error types, use `TRIGger<m>:RFFE:ERRor:BP`, `TRIGger<m>:RFFE:ERRor:LENGth`, `TRIGger<m>:RFFE:ERRor:NOResponse`, `TRIGger<m>:RFFE:ERRor:PARity`, `TRIGger<m>:RFFE:ERRor:SSC`, and `TRIGger<m>:RFFE:ERRor:USEQuence`.

*RST: STARt

To specify additional trigger cosnditions for command sequences, use the following commands:

Slave address /MID	TRIGger<m>:RFFE:SADD:CONDition TRIGger<m>:RFFE:SADD:MIN TRIGger<m>:RFFE:SADD:MAX
Register address	TRIGger<m>:RFFE:ADDRess:CONDition TRIGger<m>:RFFE:ADDRess:MIN TRIGger<m>:RFFE:ADDRess:MAX
Data pattern	TRIGger<m>:RFFE:DATA:DCON TRIGger<m>:RFFE:DATA:DMIN TRIGger<m>:RFFE:DATA:DMAX
Mask address	TRIGger<m>:RFFE:MASK:CONDition TRIGger<m>:RFFE:MASK:MAX TRIGger<m>:RFFE:MASK:MIN
Byte count	TRIGger<m>:RFFE:BCOunt:CONDition TRIGger<m>:RFFE:BCOunt:MIN TRIGger<m>:RFFE:BCOunt:MAX
Data index	TRIGger<m>:RFFE:DATA:ICONdition TRIGger<m>:RFFE:DATA:IMIN TRIGger<m>:RFFE:DATA:IMAX

TRIGger<m>:RFFE:ADDRess:CONDition <AddressOperator>

Sets the operator to trigger on a specific address pattern or a range.

Parameters:

<AddressOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one pattern to be set with TRIGger<m>:RFFE:ADDRess:MIN.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with TRIGger<m>:RFFE:ADDRess:MIN and TRIGger<m>:RFFE:ADDRess:MAX.

*RST: EQUal

TRIGger<m>:RFFE:ADDRess:MIN <AddressMin>

Specifies a pattern, or sets the start value of a pattern range.

Parameters:

<AddressMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:RFFE:ADDRess:MAX <AddressMax>

Sets the end value of a range, if the data condition is set to `INRange` or `OORange`. See `TRIGger<m>:RFFE:ADDRess:CONDition`.

Parameters:

<AddressMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:RFFE:BCOunt:CONDition <ByteCntOperator>

Sets the operator to trigger on a specific byte count or a byte count range.

Parameters:

<ByteCntOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:RFFE:BCOunt:MIN` command.

INRange | OORange
In range / Out of range: Set the minimum and maximum value using the corresponding `TRIGger<m>:RFFE:BCOunt:MIN` and `TRIGger<m>:RFFE:BCOunt:MAX`

*RST: EQUal

TRIGger<m>:RFFE:BCOunt:MIN <ByteCountMin>

Specifies a pattern, or sets the start value of a pattern range.

Parameters:

<ByteCountMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:RFFE:BCOunt:MAX <ByteCountMax>

Sets the end value of a range, if the byte/bit count condition is set to `INRange` or `OORange`. See `TRIGger<m>:RFFE:BCOunt:CONDition`.

Parameters:

<ByteCountMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:RFFE:DATA:DCON <DatPattOptor>

Sets the operator to trigger on a specific pattern or a range.

Parameters:

<DatPattOptor> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:RFFE:DATA:DMIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:RFFE:DATA:DMIN` and `TRIGger<m>:RFFE:DATA:DMAX`

*RST: EQUal

Example: See "Triggering on RFFE bus" on page 1274.

TRIGger<m>:RFFE:DATA:DMIN <DataPatternMin>

Specifies a pattern, or sets the start value of a pattern range.

Parameters:

<DataPatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

Example: See "Triggering on RFFE bus" on page 1274.

TRIGger<m>:RFFE:DATA:DMAX <DataPatternMax>

Sets the end value of a range, if the data condition is set to `INRange` or `OORange`. See `TRIGger<m>:RFFE:DATA:DCON`.

Parameters:

<DataPatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

Example: See "Triggering on RFFE bus" on page 1274.

TRIGger<m>:RFFE:DATA:ICONdition <DataIdxOperator>

Sets the operator to define the data frames in which the data pattern is expected.

Parameters:

<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | LTHan | LETHan | GTHan | GETHan
Equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set using `TRIGger<m>:RFFE:DATA:IMIN`.

INRange = RANGe
In range: Set the minimum and maximum value using `TRIGger<m>:RFFE:DATA:IMIN` and `TRIGger<m>:RFFE:DATA:IMAX`.

*RST: INRange

TRIGger<m>:RFFE:DATA:IMIN <DataIndexMin>

Specifies a pattern, or sets the start value of the index range.

Parameters:

<DataIndexMin>
Range: 1 to 32
Increment: 1
*RST: 1

TRIGger<m>:RFFE:DATA:IMAX <DataIndexMax>

Sets the end value of a range, if the index condition is set to `INRange` or `OORange`. See `TRIGger<m>:RFFE:DATA:ICONdition`.

Parameters:

<DataIndexMax>
Range: 1 to 32
Increment: 1
*RST: 32

TRIGger<m>:RFFE:SADD:CONDition <SIDOperator>

Sets the operator to trigger on a specific slave address/MID pattern or a range.

Parameters:

<SIDOperator>
EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:RFFE:SADD:MIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:RFFE:SADD:MIN` and `TRIGger<m>:RFFE:SADD:MAX` on page 2067.

*RST: EQUal

TRIGger<m>:RFFE:SADD:MIN <SIDMin>

Specifies a pattern, or sets the start value of a pattern range.

Parameters:

<SIDMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:RFFE:SADD:MAX <SIDMax>

Sets the end value of a range, if the slave address/MID condition is set to `INRange` or `OORange`. See `SEARch:TRIGger:RFFE:SADD:CONDition`.

Parameters:

<SIDMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:RFFE:MASK:CONDition <MaskOperator>

Sets the operator to trigger on a mask pattern or a range.

Parameters:

<MaskOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater than, Greater or equal than. These conditions require one pattern to be set with `TRIGger<m>:RFFE:MASK:MIN`.

INRange | OORange
In range / Out of range: Set the minimum and maximum value of the range with `TRIGger<m>:RFFE:MASK:MIN` and `TRIGger<m>:RFFE:MASK:MAX`.

*RST: EQUal

TRIGger<m>:RFFE:MASK:MAX <MaskMax>

Sets the end value of a range, if themask condition is set to `INRange` or `OORange`. See `TRIGger<m>:RFFE:MASK:CONDition`.

Parameters:

<MaskMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:RFFE:MASK:MIN <MaskMin>

Specifies a mask pattern, or sets the start value of a mask pattern range.

Parameters:
<MaskMin>

TRIGger<m>:RFFE:INTerrupt <Interrupt>

Sets the interrupt pattern for the interupt summary and notification trigger type.

Parameters:
<Interrupt>

TRIGger<m>:RFFE:ERRor:BP <BusParkError>

Enables the trigger on a bus park error - when an incorrect bus parlk cycle has been found.

Parameters:
<BusParkError> ON | OFF
*RST: ON

TRIGger<m>:RFFE:ERRor:LENGth <LengthError>

Enables the trigger on a length error - when an incorrect length of the command sequence has been found.

Parameters:
<LengthError> ON | OFF
*RST: ON

TRIGger<m>:RFFE:ERRor:NOResponse <NoResponse>

Enables the trigger on any No Response Frame.

Parameters:
<NoResponse> ON | OFF
*RST: ON

TRIGger<m>:RFFE:ERRor:PARity <ParityError>

Enables the trigger on incorrect parity bits.

Parameters:
<ParityError> ON | OFF
*RST: ON

TRIGger<m>:RFFE:ERRor:SSC <SSCError>

Enables the trigger on an SSC error - when no valid SSC sequence has been found after the idle time.

Parameters:
<SSCError> ON | OFF
*RST: ON

TRIGger<m>:RFFE:ERRor:USEQuence <UnknownSequence>

Enables the trigger on an unknown sequence - when the instrument cannot detect any supported command sequence.

Parameters:
<UnknownSequence> ON | OFF
*RST: ON

TRIGger<m>:RFFE:ERRor:GAP <GapError>

Enables the trigger on a gap error.

Parameters:
<GapError> ON | OFF
*RST: ON

TRIGger<m>:RFFE:ERRor:VERSion <VersionError>

Enables the trigger on RFFE version error.

Parameters:
<VersionError> ON | OFF
*RST: ON

TRIGger<m>:RFFE:INTerrupt <Interrupt>

Sets the interrupt pattern for the interupt summary and notification trigger type.

Parameters:
<Interrupt>

24.17.18.3 Decode results

In all BUS<m>:RFFE commands, the suffix <m> selects the serial bus. Suffix <n> selects index of the command sequence.

To load and acivate a label list, use:

- `BUS<m>:NEWList` on page 1672
- `BUS<m>:SYMBols` on page 1672

BUS<m>:RFFE:FCOunt?....2070
BUS<m>:RFFE:SEQuence<n>:STATe?....2070
BUS<m>:RFFE:SEQuence<n>:STARt?....2071
BUS<m>:RFFE:SEQuence<n>:STOP?....2071

BUS<m>:RFFE:SEQuence<n>:TYPE?..... 2071
BUS<m>:RFFE:SEQuence<n>:SADD?..... 2072
BUS<m>:RFFE:SEQuence<n>:ADDRess?..... 2072
BUS<m>:RFFE:SEQuence<n>:BCOunt?..... 2072
BUS<m>:RFFE:SEQuence<n>:DATA?..... 2073
BUS<m>:RFFE:SEQuence<n>:SYMBol?..... 2073
BUS<m>:RFFE:SEQuence<n>:PADone?..... 2073
BUS<m>:RFFE:SEQuence<n>:PADZero?..... 2073
BUS<m>:RFFE:SEQuence<n>:PCTRl?..... 2074
BUS<m>:RFFE:SEQuence<n>:RBTRate?..... 2074
BUS<m>:RFFE:SEQuence<n>:WBTRate?..... 2074
BUS<m>:RFFE:SEQuence<n>:BYTE<o>:NAME?..... 2075
BUS<m>:RFFE:SEQuence<n>:BYTE<o>:STATe?..... 2075
BUS<m>:RFFE:SEQuence<n>:BYTE<o>:VALue?..... 2075

BUS<m>:RFFE:FCOunt?

Returns the number of command sequences in the current acquisition.

Suffix:

<m> 1..4

Return values:

<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:STATe?

Returns the overall state of the selected command sequence.

Suffix:

<m> 1..4

<n> *

Return values:

<FrameState> OK | SSC | VERSion | PARity | LENGth | BPERR | GAP | NORESPONSE | INComplete

`OK`: the sequence is valid.
`SSC`: SSC error.
`VERSion`: RFFE version.
`PARity`: parity error.
`LENGth`: length error.
`BPERR`: bus park error.
`NORESPONSE`: No response.
`INComplete`: The sequence is not completely contained in the acquisition. The acquired part of the sequence is valid.

*RST: OK

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:STARt?
BUS<m>:RFFE:SEQuence<n>:STOP?

Return the start time (SSC) and stop time (BP) of the selected command sequence.

Suffix:
<m> 1..4

<n> *

Return values:
<Start>, <Stop> Time

Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:TYPE?

Returns the type of the selected command sequence.

Suffix:
<m> 1..4

<n> *

Return values:
<FrameType> UNDESC | RZWR | RWR | RRD | ERWR | ERRD | ERWL | ERRL | MASR | MASW | MASOHO | IRSUM | MSKW | MCTR | MCTW | UNDEF | ERRor

`RZWR`: Register 0 Write
`RWR`: Register Write
`RRD`: Register Read
`ERWR`: Extended Register Write
`ERRD`: Extended Register Read
`ERWL`: Extended Register Write Long
`ERRL`: Extended Register Read Long
`MASR`: Master Read
`MASW`: Master Write
`MASOHO`: Master Ownership Handover
`IRSUM`: Interrupt Summary and Notification
`MSKW`: Masked Write
`MCTR`: Master Context Transfer Read
`MCTW`: Master Context Transfer Write
`ERRor`: The bits defining the command sequence are not valid, no supported command sequence

*RST: RZWR

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:SADD?

Returns the slave address of the selected command sequence.

Suffix:
<m> 1..4

<n> *

Return values:
<FrameSID> Range: 0 to 15
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:ADDRess?

Returns the register address of the selected command sequence.

Suffix:
<m> 1..4

<n> *

Return values:
<FrameAddress> Range: Depends on the sequence type, address can have 0, 5, 8, or 16 bits
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:BCOunt?

Returns the bit/ byte count of the selected command sequence.

Suffix:
<m> 1..4

<n> *

Return values:
<ByteCount> Decimal value
Range: 15
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:DATA?

Returns the data bytes of the specified command sequence.

Use `BUS<m>:RFFE:SEQuence<n>:BYTE<o>:VALue?` to access the sequence bytes.

Suffix:

<m>	1..4
<n>	*

Return values:

<Data>	Bit pattern (B1, B2, B3, B4…) in variable length up to eight bytes.

Example:

```
BUS:RFFE:SEQuence3:DATA?
<-- #H08,#H49,#H54,#H33,#HFF
```

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:SYMBol?

Returns the symbolic label that belongs to the address of the selected command sequence.

Suffix:

<m>	1..4
<n>	*

Return values:

<Translation>	String containing the label name

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:PADone?

Returns the address-one parity for the selected command sequence.

Suffix:

<m>	1..4
<n>	*

Return values:

<ParityAddr1>	OK | PARity `PARity`: PArrity error *RST: OK

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:PADZero?

Returns the address-zero parity for the selected command sequence.

Suffix:
<m> 1..4

<n> *

Return values:
<ParityAddr0> OK | PARity

`PARity`: PArrity error

*RST: OK

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:PCTRl?

Returns the control parity for the selected command sequence.

Suffix:
<m> 1..4

<n> *

Return values:
<ParityCtrl> OK | PARity

`PARity`: PArrity error

*RST: OK

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:RBTRate?

Returns the read bit rate of the selected command sequence.

Suffix:
<m> 1..4

<n> *

Return values:
<ReadBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:WBTRate?

Returns the write bit rate of the selected command sequence.

Suffix:
<m> 1..4

<n> *

Return values:

<WriteBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:BYTE<o>:NAME?

Returns the name of the specified data byte.

Suffix:

<m> 1..4

<n> *

<o> *
Selects the number of the data byte (1...n).

Return values:

<FrameByteName>

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:BYTE<o>:STATe?

Returns the state of the specified data byte.

Suffix:

<m> 1..4

<n> *

<o> *
Selects the number of the data byte (1...n).

Return values:

<ByteState> OK | PARity

OK: the byte is valid.
PARity: parity error

*RST: OK

Usage: Query only

BUS<m>:RFFE:SEQuence<n>:BYTE<o>:VALue?

Returns the value of the specified byte in the specified command sequence.

Suffix:

<m> 1..4

<n> *

<o> *
Selects the number of the data byte (1...n).

Return values:

<ByteValue> To set the value format, use `FORMat:BPATtern`.
The values below – range, increment and reset – are decimal values.

Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

24.17.18.4 Search settings

For programming example, see "Searching RFFE data" on page 1274.

SEARch:TRIGger:RFFE:TYPE....2076
SEARch:TRIGger:RFFE:SADD:CONDition....2078
SEARch:TRIGger:RFFE:ADDRess:CONDition....2078
SEARch:TRIGger:RFFE:BCOunt:CONDition....2079
SEARch:TRIGger:RFFE:DATA:DCON....2079
SEARch:TRIGger:RFFE:DATA:ICONdition....2079
SEARch:TRIGger:RFFE:SADD:MIN....2080
SEARch:TRIGger:RFFE:ADDRess:MIN....2080
SEARch:TRIGger:RFFE:BCOunt:MIN....2080
SEARch:TRIGger:RFFE:DATA:IMIN....2080
SEARch:TRIGger:RFFE:DATA:DMIN....2080
SEARch:TRIGger:RFFE:SADD:MAX....2080
SEARch:TRIGger:RFFE:ADDRess:MAX....2080
SEARch:TRIGger:RFFE:BCOunt:MAX....2080
SEARch:TRIGger:RFFE:DATA:IMAX....2080
SEARch:TRIGger:RFFE:DATA:DMAX....2080
SEARch:TRIGger:RFFE:INTerrupt....2080
SEARch:TRIGger:RFFE:ERRor:LENGth....2081
SEARch:TRIGger:RFFE:ERRor:PARity....2081
SEARch:TRIGger:RFFE:ERRor:SSC....2081
SEARch:TRIGger:RFFE:ERRor:GAP....2082
SEARch:TRIGger:RFFE:ERRor:VERSion....2082

SEARch:TRIGger:RFFE:TYPE <SearchName>,<Type>
SEARch:TRIGger:RFFE:TYPE? <SearchName>

Sets the event to be searched for.

See also: "Type" on page 771

Parameters:

<Type> STARt | STOP | RZWR | RWR | RRD | ERWR | ERRD | ERWL | ERRL | IRSUM | MSKW | MASOHO | MASW | MASR | MCTW | MCTR | ERRor

STARt
Start of a command sequence (SSC). Optional: slave address.

STOP
End of a command sequence (start of the bus park cycle). Optional: slave address condition.

RZWR
Register 0 Write command sequence. Optional: slave address condition and data pattern condition.

RWR
Register Write command sequence. Optional: slave address condition, register address condition, and data pattern condition.

RRD
Register Read command sequence. Optional: slave address condition, register address condition, and data pattern condition.

ERWR
Extended Register Write command sequences. Optional: slave address condition, register address condition, byte count condition, data index condition, and data pattern condition.

ERRD
Extended Register Read command sequence. Optional search conditions are the same as for `ERWR`.

ERWL
Extended Register Write Long command sequence. Optional search conditions are the same as for `ERWR`.

ERRL
Extended Register Read Long command sequence. Optional search conditions are the same as for `ERWR`.

MASR
Master Read command sequence. Optional: slave address condition, register address condition, and data pattern condition.

MASW
Master Write command sequence. Optional: slave address condition, register address condition, and data pattern condition.

MASOHO
Master Ownership Handover command sequence. Optional: slave address condition, and data pattern condition.

MSKW
Masked Write command sequence. Optional: slave address condition, register address condition, mask condition and data pattern condition.

MCTW
Master Context Transfer Write command sequence. Optional: MID condition, register address condition, byte count condition, data index condition, and data pattern condition.

MCTR
Master Context Transfer Read command sequence. Optional search conditions are the same as for `MCTW`

IRSUM
Interrupt Summary and Notification. Optional: slave address condition, byte count condition, and data pattern condition.

ERRor
Searches for enabled errors. To enable the error types, use:
`SEARch:TRIGger:RFFE:ERRor:LENGth`
`SEARch:TRIGger:RFFE:ERRor:PARity`
`SEARch:TRIGger:RFFE:ERRor:SSC`

*RST: STARt

Parameters for setting and query:
<SearchName> String parameter

Example: See "Searching RFFE data" on page 1274

To specify additional search conditions for command sequences, use the following commands:

Slave address	`SEARch:TRIGger:RFFE:SADD:CONDition`
	`SEARch:TRIGger:RFFE:SADD:MIN`
	`SEARch:TRIGger:RFFE:SADD:MAX`

Register address	`SEARch:TRIGger:RFFE:ADDRess:CONDition`
	`SEARch:TRIGger:RFFE:ADDRess:MIN`
	`SEARch:TRIGger:RFFE:ADDRess:MAX`

Data pattern	`SEARch:TRIGger:RFFE:DATA:DCON`
	`SEARch:TRIGger:RFFE:DATA:DMIN`
	`SEARch:TRIGger:RFFE:DATA:DMAX`

Byte count	`SEARch:TRIGger:RFFE:BCOunt:CONDition`
	`SEARch:TRIGger:RFFE:BCOunt:MIN`
	`SEARch:TRIGger:RFFE:BCOunt:MAX`

Data index	`SEARch:TRIGger:RFFE:DATA:ICONdition`
	`SEARch:TRIGger:RFFE:DATA:IMIN`
	`SEARch:TRIGger:RFFE:DATA:IMAX`

SEARch:TRIGger:RFFE:SADD:CONDition <SearchName>,<SIDOperator>
SEARch:TRIGger:RFFE:SADD:CONDition? <SearchName>
SEARch:TRIGger:RFFE:ADDRess:CONDition <SearchName>,<AddressOperator>
SEARch:TRIGger:RFFE:ADDRess:CONDition? <SearchName>

SEARch:TRIGger:RFFE:BCOunt:CONDition <SearchName>,<ByteCntOperator>
SEARch:TRIGger:RFFE:BCOunt:CONDition? <SearchName>
SEARch:TRIGger:RFFE:DATA:DCON <SearchName>,<DatPattOptor>
SEARch:TRIGger:RFFE:DATA:DCON? <SearchName>

Sets the operator to search for a specific pattern or a range

Parameters:

<DatPattOptor> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set using the corresponding `SEARch:TRIGger:RFFE:...:MIN` command.

INRange | OORange
In range / Out of range: Set the minimum and maximum value using the corresponding `SEARch:TRIGger:RFFE:...:MIN` and `SEARch:TRIGger:RFFE:...:MAX`

*RST: EQUal

Parameters for setting and query:

<SearchName> String parameter

Example: See "Searching RFFE data" on page 1274

SEARch:TRIGger:RFFE:DATA:ICONdition <SearchName>,<DataIdxOperator>
SEARch:TRIGger:RFFE:DATA:ICONdition? <SearchName>

Sets the operator to define the data frames in which the data pattern is searched

Parameters:

<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | LTHan | LETHan | GTHan | GETHan
Equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set using `SEARch:TRIGger:RFFE:DATA:IMIN`.

INRange = RANGe
In range: Set the minimum and maximum value using `SEARch:TRIGger:RFFE:DATA:IMIN` and `SEARch:TRIGger:RFFE:DATA:IMAX`.

*RST: INRange

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:RFFE:SADD:MIN <SearchName>,<SIDMin>
SEARch:TRIGger:RFFE:SADD:MIN? <SearchName>
SEARch:TRIGger:RFFE:ADDRess:MIN <SearchName>,<AddressMin>
SEARch:TRIGger:RFFE:ADDRess:MIN? <SearchName>
SEARch:TRIGger:RFFE:BCOunt:MIN <SearchName>,<ByteCountMin>
SEARch:TRIGger:RFFE:BCOunt:MIN? <SearchName>
SEARch:TRIGger:RFFE:DATA:IMIN <SearchName>,<DataIndexMin>
SEARch:TRIGger:RFFE:DATA:IMIN? <SearchName>
SEARch:TRIGger:RFFE:DATA:DMIN <SearchName>,<DataPatternMin>
SEARch:TRIGger:RFFE:DATA:DMIN? <SearchName>

Specifies a pattern, or sets the the start value of a pattern range.

Parameters:

<DataPatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

Example: See "Searching RFFE data" on page 1274

SEARch:TRIGger:RFFE:SADD:MAX <SearchName>,<SIDMax>
SEARch:TRIGger:RFFE:SADD:MAX? <SearchName>
SEARch:TRIGger:RFFE:ADDRess:MAX <SearchName>,<AddressMax>
SEARch:TRIGger:RFFE:ADDRess:MAX? <SearchName>
SEARch:TRIGger:RFFE:BCOunt:MAX <SearchName>,<ByteCountMax>
SEARch:TRIGger:RFFE:BCOunt:MAX? <SearchName>
SEARch:TRIGger:RFFE:DATA:IMAX <SearchName>,<DataIndexMax>
SEARch:TRIGger:RFFE:DATA:IMAX? <SearchName>
SEARch:TRIGger:RFFE:DATA:DMAX <SearchName>,<DataPatternMax>
SEARch:TRIGger:RFFE:DATA:DMAX? <SearchName>

Sets the the end value of an range if the corresponding condition is set to `INRange` or `OORange`. See `SEARch:TRIGger:RFFE:...:CON`

Parameters:

<DataPatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName> String parameter

SEARch:TRIGger:RFFE:INTerrupt <SearchName>,<Interrupt>
SEARch:TRIGger:RFFE:INTerrupt? <SearchName>

Defines the pattern of the interrrupt identification sequence, which consists of interrupt slots 15 to 0.

Parameters:
<Interrupt> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName> String parameter

Firmware/software: FW 3.30

SEARch:TRIGger:RFFE:ERRor:LENGth <SearchName>,<LengthError>
SEARch:TRIGger:RFFE:ERRor:LENGth? <SearchName>

Enables the search for length errors - when an incorrect length of the command sequence has been found.

Parameters:
<LengthError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName> String parameter

SEARch:TRIGger:RFFE:ERRor:PARity <SearchName>,<ParityError>
SEARch:TRIGger:RFFE:ERRor:PARity? <SearchName>

Enables the search for incorrect parity bits.

Parameters:
<ParityError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName> String parameter

SEARch:TRIGger:RFFE:ERRor:SSC <SearchName>,<SSCError>
SEARch:TRIGger:RFFE:ERRor:SSC? <SearchName>

Enables the search for SSC errors - when no valid SSC sequence has been found after the idle time.

Parameters:
<SSCError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName> String parameter

SEARch:TRIGger:RFFE:ERRor:GAP <SearchName>,<GapError>
SEARch:TRIGger:RFFE:ERRor:GAP? <SearchName>

Enables the search for RFFE gap time error.

Parameters:
<GapError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:RFFE:ERRor:VERSion <SearchName>,<VersionError>
SEARch:TRIGger:RFFE:ERRor:VERSion? <SearchName>

Enables the search for RFFE vresion error.

Parameters:
<VersionError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

24.17.18.5 Search results

The search on decoded RFFE data returns the same results as the queries for decode results.

In all SEARch:RESult:RFFE commands, the suffix <m> selects the command sequence. Suffix <n> selects index of byte inside a command sequence.

In search result commands, you have to specify the <SearchName> parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 24.17.18.3, "Decode results", on page 2069.

For programming example, see "Searching RFFE data" on page 1274.

SEARch:RESult:RFFE:FCOunt?....2083
SEARch:RESult:RFFE:SEQuence<m>:TYPE?....2083
SEARch:RESult:RFFE:SEQuence<m>:STATe?....2083
SEARch:RESult:RFFE:SEQuence<m>:STARt?....2083
SEARch:RESult:RFFE:SEQuence<m>:SADD?....2084
SEARch:RESult:RFFE:SEQuence<m>:STOP?....2084
SEARch:RESult:RFFE:SEQuence<m>:ADDRess?....2084
SEARch:RESult:RFFE:SEQuence<m>:BCOunt?....2085
SEARch:RESult:RFFE:SEQuence<m>:DATA?....2085
SEARch:RESult:RFFE:SEQuence<m>:SYMBol?....2085
SEARch:RESult:RFFE:SEQuence<m>:BYTE<n>:STATe?....2085
SEARch:RESult:RFFE:SEQuence<m>:BYTE<n>:VALue?....2086

SEARch:RESult:RFFE:FCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:RFFE:SEQuence<m>:TYPE? <SearchName>

Returns the type of the specified command sequence.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameType> UNDESC | RZWR | RWR | RRD | ERWR | ERRD | ERWL | ERRL | MASR | MASW | MASOHO | IRSUM | MSKW | MCTR | MCTW | UNDEF | ERRor

See `SEARch:TRIGger:RFFE:TYPE`.

*RST: RZWR

Usage: Query only

SEARch:RESult:RFFE:SEQuence<m>:STATe? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameState> OK | SSC | VERSion | PARity | LENGth | BPERR | GAP | NORESPONSE | INComplete

*RST: OK

Usage: Query only

SEARch:RESult:RFFE:SEQuence<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<FrameStart> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:RFFE:SEQuence<m>:SADD? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<FrameSID> Range: 0 to 15
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:RFFE:SEQuence<m>:STOP? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<FrameStop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:RFFE:SEQuence<m>:ADDRess? <SearchName>

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<FrameAddress> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:RFFE:SEQuence<m>:BCOunt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<ByteCount> Range: 0 to 256
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:RFFE:SEQuence<m>:DATA? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:RFFE:SEQuence<m>:SYMBol? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Translation>

Usage: Query only

SEARch:RESult:RFFE:SEQuence<m>:BYTE<n>:STATe? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<ByteState> OK | PARity
OK: the byte is valid.
PARity: parity error
*RST: OK

Usage: Query only

SEARch:RESult:RFFE:SEQuence<m>:BYTE<n>:VALue? <SearchName>

Suffix:
<m> *
<n> *

Query parameters:
<SearchName>

Return values:
<ByteValue> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

24.17.19 Ethernet 100BASE-T1 (option R&S RTO6-K560)

- Configuration....2086
- Trigger....2090
- Decode results....2097
- Search settings....2103
- Search results....2111

24.17.19.1 Configuration

In all `BUS<m>:HBTO` commands, the suffix <m> selects the serial bus.

BUS<m>:HBTO:ATTN....2087
BUS<m>:HBTO:FDAM....2087
BUS<m>:HBTO:FDAP....2087
BUS<m>:HBTO:FDIF....2087
BUS<m>:HBTO:FTYP....2088
BUS<m>:HBTO:RDAM....2088
BUS<m>:HBTO:RDAP....2088
BUS<m>:HBTO:RDIF....2088
BUS<m>:HBTO:RTYP....2089
BUS<m>:HBTO:THReshold....2089
BUS<m>:HBTO:POLarity....2089
BUS<m>:HBTO:MODE....2089
BUS<m>:HBTO:SYMRate....2089

BUS<m>:HBTO:ATTN <Attenuation>

Sets the attenuation factor. It is used to de-amplify the reverse signal before subtracting it from the forward signal.

Suffix:
<m> 1..4

Parameters:
<Attenuation> Range: -100 to 0
Increment: 0.1
*RST: -26
Default unit: dB

BUS<m>:HBTO:FDAM <SourceDAminus>

Selects the DA- source of the provided forward single ended signal, if `BUS<m>:HBTO:FTYP` is set to `SINGle`.

Suffix:
<m> 1..4

Parameters:
<SourceDAminus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4
*RST: C2W1

BUS<m>:HBTO:FDAP <SourceDAplus>

Selects the DA+ source of the provided forward single ended signal, if `BUS<m>:HBTO:FTYP` is set to `SINGle`.

Suffix:
<m> 1..4

Parameters:
<SourceDAplus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4
*RST: C1W1

BUS<m>:HBTO:FDIF <SrcDiff>

Sets the source of the provided forward differential signal, if `BUS<m>:HBTO:FTYP` is set to `DIFFerential`.

Suffix:
<m> 1..4

Parameters:

<SrcDiff> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: C1W1

BUS<m>:HBTO:FTYP <SignalType>

Sets the type of forward signal measurement.

Suffix:

<m> 1..4

Parameters:

<SignalType> DIFFerential | SINGle

*RST: DIFFerential

BUS<m>:HBTO:RDAM <SrcRevDAminus>

Selects the DA- source of the provided reversed single ended signal, if `BUS<m>:HBTO:RTYP` is set to `SINGle`.

Suffix:

<m> 1..4

Parameters:

<SrcRevDAminus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

BUS<m>:HBTO:RDAP <SourceRevDAplus>

Selects the DA+ source of the provided reversed single ended signal, if `BUS<m>:HBTO:RTYP` is set to `SINGle`.

Suffix:

<m> 1..4

Parameters:

<SourceRevDAplus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

BUS<m>:HBTO:RDIF <SrcRevDiff>

Sets the source of the provided reversed differential signal, if `BUS<m>:HBTO:RTYP` is set to `DIFFerential`.

Suffix:

<m> 1..4

Parameters:

<SrcRevDiff> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

BUS<m>:HBTO:RTYP <SignalRevType>

Sets the type of reverse signal measurement.

Suffix:
<m> 1..4

Parameters:
<SignalRevType> DISabled | DIFFerential | SINGle
*RST: DISabled

BUS<m>:HBTO:THReshold <ThresholdUpper>

Sets an user-defined threshold value.

Suffix:
<m> 1..4

Parameters:
<ThresholdUpper> Range: 0 to 2
Increment: 1E-3
*RST: 0.4
Default unit: V

BUS<m>:HBTO:POLarity <Polarity>

Selects the polarity of the data signal.

Suffix:
<m> 1..4

Parameters:
<Polarity> NORMal | INVert
*RST: NORMal

BUS<m>:HBTO:MODE <Mode>

Selects the direction of the full-duplex signal you want to look at.

Suffix:
<m> 1..4

Parameters:
<Mode> MASTer | SLAVe | AUTO
*RST: AUTO

BUS<m>:HBTO:SYMRate <SymbolRate>

Defines the rate of ternary symbols.

Suffix:
<m> 1..4

Parameters:

<SymbolRate>	Range:	10 to 150
	Increment:	1E-3
	*RST:	66.6667
	Default unit:	MSymb/s

24.17.19.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:HBTO:TYPE 2090
TRIGger<m>:HBTO:DADDress:CONDition 2091
TRIGger<m>:HBTO:DADDress:MIN 2091
TRIGger<m>:HBTO:DADDress:MAX 2092
TRIGger<m>:HBTO:SADDress:CONDition 2092
TRIGger<m>:HBTO:SADDress:MIN 2092
TRIGger<m>:HBTO:SADDress:MAX 2093
TRIGger<m>:HBTO:LENGth:CONDition 2093
TRIGger<m>:HBTO:LENGth:MIN 2093
TRIGger<m>:HBTO:LENGth:MAX 2093
TRIGger<m>:HBTO:CRC:CONDition 2094
TRIGger<m>:HBTO:CRC:MIN 2094
TRIGger<m>:HBTO:CRC:MAX 2094
TRIGger<m>:HBTO:DATA:DCONdition 2095
TRIGger<m>:HBTO:DATA:DMIN 2095
TRIGger<m>:HBTO:DATA:DMAX 2095
TRIGger<m>:HBTO:DATA:ICONdition 2096
TRIGger<m>:HBTO:DATA:IMIN 2096
TRIGger<m>:HBTO:DATA:IMAX 2096
TRIGger<m>:HBTO:ERRor:PREamble 2096
TRIGger<m>:HBTO:ERRor:CRC 2097
TRIGger<m>:HBTO:ERRor:SFD 2097

TRIGger<m>:HBTO:TYPE <Type>

Selects the type of frame to be triggered on.

Suffix:

<m>	1..3

Parameters:

<Type>	STARt | MAC | IDLE | ERRor

STARt
Start of frame.

MAC
MAC frame. This frame contains information that define how to go about transmitting and receiving frames.

IDLE
IDLE frame. This frame is used for clock synchronization.

ERRor
Error frame. Thi frame contains erroneous bits.

*RST: STARt

TRIGger<m>:HBTO:DADDress:CONDition <DestAddrOptor>

Sets the condition for the destination address. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DestAddrOptor> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:HBTO:DADDress:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:HBTO:DADDress:MIN` and `TRIGger<m>:HBTO:DADDress:MAX`.

*RST: EQUal

TRIGger<m>:HBTO:DADDress:MIN <DestAddrPattMin>

Specifies the destination address bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DestAddrPattMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:DADDress:MAX <DestAddrPattMax>

Sets the end value of the destination address range if `TRIGger<m>:HBTO:DADDress:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<DestAddrPattMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:SADDress:CONDition <SrcAddrOperator>

Sets the condition for the source address. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<SrcAddrOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:HBTO:SADDress:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:HBTO:SADDress:MIN` and `TRIGger<m>:HBTO:SADDress:MAX`.

*RST: EQUal

TRIGger<m>:HBTO:SADDress:MIN <SrcAddrPattMin>

Specifies the source address bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<SrcAddrPattMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:SADDress:MAX <SrcAddrPattMax>

Sets the end value of the source address range if `TRIGger<m>:HBTO:SADDress:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<SrcAddrPattMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:LENGth:CONDition <TypeOperator>

Sets the condition for the length / type. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<TypeOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:HBTO:LENGth:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:HBTO:LENGth:MIN` and `TRIGger<m>:HBTO:LENGth:MAX`.

*RST: EQUal

TRIGger<m>:HBTO:LENGth:MIN <TypePatternMin>

Specifies the length / type bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<TypePatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:LENGth:MAX <TypePatternMax>

Sets the end value of the length / type range if `TRIGger<m>:HBTO:LENGth:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<TypePatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:CRC:CONDition <CRCOperator>

Sets the condition for the frame check. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<CRCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:HBTO:CRC:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:HBTO:CRC:MIN` and `TRIGger<m>:HBTO:CRC:MAX`.

*RST: EQUal

TRIGger<m>:HBTO:CRC:MIN <CRCPatternMin>

Specifies the frame check bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<CRCPatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:CRC:MAX <CRCPatternMax>

Sets the end value of the frame check range if `TRIGger<m>:HBTO:CRC:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:

<CRCPatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:DATA:DCONdition <DataOperator>

Sets the condition for the data. You can define an exact value or a value range.

Suffix:

<m> 1..3

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:HBTO:DATA:DMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:HBTO:DATA:DMIN` and `TRIGger<m>:HBTO:DATA:DMAX`.

*RST: EQUal

TRIGger<m>:HBTO:DATA:DMIN <DataMin>

Specifies the data bit pattern, or sets the start value of a pattern range.

Suffix:

<m> 1..3

Parameters:

<DataMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:DATA:DMAX <DataMax>

Sets the end value of the data range if `TRIGger<m>:HBTO:DATA:DCONdition` is set to `INRange` or `OORange`.

Suffix:

<m> 1..3

Parameters:

<DataMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:HBTO:DATA:ICONdition <DataIdxOperator>

Sets the condition for the index. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:HBTO:DATA:IMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:HBTO:DATA:IMIN` and `TRIGger<m>:HBTO:DATA:IMAX`.

*RST: INRange

TRIGger<m>:HBTO:DATA:IMIN <DataIndexMin>

Specifies the index bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DataIndexMin> Range: 1 to 65535
Increment: 1
*RST: 1

TRIGger<m>:HBTO:DATA:IMAX <DataIndexMax>

Sets the end value of the index range if `TRIGger<m>:HBTO:DATA:ICONdition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<DataIndexMax> Range: 1 to 65535
Increment: 1
*RST: 0

TRIGger<m>:HBTO:ERRor:PREamble <ErrorPreamble>

Enables / disables trigger on preamble error.

Suffix:
<m> 1..3

Parameters:
<ErrorPreamble> ON | OFF
*RST: ON

TRIGger<m>:HBTO:ERRor:CRC <ErrorCRC>

Enables / disables trigger on Cyclic Redundancy Check (CRC) error.

Suffix:
<m> 1..3

Parameters:
<ErrorCRC> ON | OFF
*RST: ON

TRIGger<m>:HBTO:ERRor:SFD <ErrorSFD>

Enables / disables trigger on start frame delimiter (SFD) error.

Suffix:
<m> 1..3

Parameters:
<ErrorSFD> ON | OFF
*RST: ON

24.17.19.3 Decode results

To show the results on the screen, use the following commands:

- `BUS<m>:RESult` on page 1671

In all `BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>` commands, the suffix <m> selects the serial bus, suffix <n> selects the frame in the decode table and suffix <o> selects the word in the selected frame.

BUS<m>:HBTO:RESult:FCOunt? 2098
BUS<m>:HBTO:RESult:FRAMe<n>:FTYPe? 2098
BUS<m>:HBTO:RESult:FRAMe<n>:STATe? 2098
BUS<m>:HBTO:RESult:FRAMe<n>:STARt? 2099
BUS<m>:HBTO:RESult:FRAMe<n>:STOP? 2099
BUS<m>:HBTO:RESult:FRAMe<n>:DESTaddress? 2100
BUS<m>:HBTO:RESult:FRAMe<n>:SRCaddress? 2100
BUS<m>:HBTO:RESult:FRAMe<n>:TYPE? 2100
BUS<m>:HBTO:RESult:FRAMe<n>:DATA? 2100
BUS<m>:HBTO:RESult:FRAMe<n>:CRC? 2101
BUS<m>:HBTO:RESult:FRAMe<n>:NUMWords? 2101
BUS<m>:HBTO:RESult:FRAMe<n>:DSYMbol? 2101
BUS<m>:HBTO:RESult:FRAMe<n>:SSYMbol? 2101

BUS<m>:HBTO:RESult:FRAMe<n>:BITRate?.....2102
BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:TYPE?.....2102
BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:VALue?.....2102
BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:VSTR?.....2103

BUS<m>:HBTO:RESult:FCOunt?

Returns the number of decoded frames.

Suffix:
<m> 1..4

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:FTYPe?

Returns the type of frame for the selected frame.

Suffix:
<m> 1..4

<n> *

Return values:
<FrameType> IDLE | MAC | FILLer | UNKNown

IDLE
IDLE frame. This frame is used for clock synchronization.

MAC
MAC frame. This frame contains information that define how to go about transmitting and receiving frames.

FILLer
Filler frame. The frame is used to maintain transmission activity.

UNKNown
No meaningful frame can be determined.

*RST: MAC

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:STATe?

Returns the state of the frame.

Suffix:
<m> 1..4

<n> *

Return values:

<State> OK | ERR_PREAMBLE | ERR_SFD | ERR_LENGTH | ERR_CRC | UNCorrelated | INComplete

OK
Valid frame.

ERR_PREAMBLE
Erroneous frame due to preamble error.

ERR_SFD
Erroneous frame due to SFD error.

ERR_LENGTH
Erroneous frame due to length / type error.

ERR_CRC
Erroneous frame due to CRC error.

*RST: OK

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:STARt?

Returns the start time of the selected frame.

Suffix:

<m> 1..4

<n> *

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:STOP?

Returns the end time of the selected frame.

Suffix:

<m> 1..4

<n> *

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:DESTaddress?

Returns the destination address of the selected frame.

Suffix:
<m> 1..4

<n> *

Return values:
<DestAddress>

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:SRCaddress?

Returns the source address of the selected frame.

Suffix:
<m> 1..4

<n> *

Return values:
<SrcAddress>

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:TYPE?

Returns the value of length / type field of the selected frame.

Suffix:
<m> 1..4

<n> *

Return values:
<TypeLen> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:DATA?

Returns the data for the selected frame, corresponds to the Data column in the decode results table.

Suffix:
<m> 1..4

<n> *

Return values:
<Data>

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:CRC?

Returns the CRC (frame check) checksum of the selected frame.

Suffix:
<m> 1..4

<n> *

Return values:
<CRC> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:NUMWords?

Returns the number of decoded words for the selected frame.

Suffix:
<m> 1..4

<n> *

Return values:
<NumWords> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:DSYMbol?

Returns the destination symbols of the selected frame.

Suffix:
<m> 1..4

<n> *

Return values:
<DestTranslation>

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:SSYMbol?

Returns the source symbols of the selected frame.

Suffix:
<m> 1..4

<n> *

Return values:
<SrcTranslation>

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:BITRate?

Returns the primary bit rate.

Suffix:
<m> 1..4

<n> *

Return values:
<PrimaryBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:TYPE?

Returns the data type for the selected word.

Suffix:
<m> 1..4

<n> *

<o> *

Return values:
<WordType>

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:VALue?

Returns the data value for the selected word.

Suffix:
<m> 1..4

<n> *

<o> *

Return values:
<WordValue> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:VSTR?

Returns the string equivalent of data value for the selected cell.

Suffix:

<m>	1..4
<n>	*
<o>	*

Return values:

<WordValueString>	String parameter

Usage: Query only

24.17.19.4 Search settings

SEARch:TRIGger:HBTO:TYPE 2103
SEARch:TRIGger:HBTO:DADDress:CONDition 2104
SEARch:TRIGger:HBTO:DADDress:MIN 2104
SEARch:TRIGger:HBTO:DADDress:MAX 2105
SEARch:TRIGger:HBTO:SADDress:CONDition 2105
SEARch:TRIGger:HBTO:SADDress:MIN 2105
SEARch:TRIGger:HBTO:SADDress:MAX 2106
SEARch:TRIGger:HBTO:LENGth:CONDition 2106
SEARch:TRIGger:HBTO:LENGth:MIN 2106
SEARch:TRIGger:HBTO:LENGth:MAX 2107
SEARch:TRIGger:HBTO:CRC:CONDition 2107
SEARch:TRIGger:HBTO:CRC:MIN 2107
SEARch:TRIGger:HBTO:CRC:MAX 2108
SEARch:TRIGger:HBTO:DATA:DCONdition 2108
SEARch:TRIGger:HBTO:DATA:DMIN 2108
SEARch:TRIGger:HBTO:DATA:DMAX 2109
SEARch:TRIGger:HBTO:DATA:ICONdition 2109
SEARch:TRIGger:HBTO:DATA:IMIN 2109
SEARch:TRIGger:HBTO:DATA:IMAX 2110
SEARch:TRIGger:HBTO:ERRor:PREamble 2110
SEARch:TRIGger:HBTO:ERRor:CRC 2110
SEARch:TRIGger:HBTO:ERRor:SFD 2110

SEARch:TRIGger:HBTO:TYPE <SearchName>,<Type>
SEARch:TRIGger:HBTO:TYPE? <SearchName>

Selects the type of frame to be searched for.

Parameters:

<Type> STARt | MAC | IDLE | ERRor

STARt
Start of frame.

MAC
MAC frame. This frame contains information that define how to go about transmitting and receiving frames.

IDLE
IDLE frame. This frame is used for clock synchronization.

ERRor
Error frame. Thi frame contains erroneous bits.

*RST: STARt

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:DADDress:CONDition <SearchName>,<DestAddrOptor>
SEARch:TRIGger:HBTO:DADDress:CONDition? <SearchName>

Sets the condition for the destination address. You can define an exact value or a value range.

Parameters:
<DestAddrOptor> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:HBTO:DADDress:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:HBTO:DADDress:MIN` and `SEARch:TRIGger:HBTO:DADDress:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:DADDress:MIN <SearchName>,<DestAddrPattMin>
SEARch:TRIGger:HBTO:DADDress:MIN? <SearchName>

Specifies the destination address bit pattern, or sets the start value of a pattern range.

Parameters:
<DestAddrPattMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:DADDress:MAX <SearchName>,<DestAddrPattMax>
SEARch:TRIGger:HBTO:DADDress:MAX? <SearchName>

Sets the end value of the destination address range if `SEARch:TRIGger:HBTO:DADDress:CONDition` is set to `INRange` or `OORange`.

Parameters:

<DestAddrPattMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:SADDress:CONDition <SearchName>,<SrcAddrOperator>
SEARch:TRIGger:HBTO:SADDress:CONDition? <SearchName>

Sets the condition for the source address. You can define an exact value or a value range.

Parameters:

<SrcAddrOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:HBTO:SADDress:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:HBTO:SADDress:MIN` and `SEARch:TRIGger:HBTO:SADDress:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:SADDress:MIN <SearchName>,<SrcAddrPattMin>
SEARch:TRIGger:HBTO:SADDress:MIN? <SearchName>

Specifies the source address bit pattern, or sets the start value of a pattern range.

Parameters:

<SrcAddrPattMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:SADDress:MAX <SearchName>,<SrcAddrPattMax>
SEARch:TRIGger:HBTO:SADDress:MAX? <SearchName>

Sets the end value of the source address range if `SEARch:TRIGger:HBTO:SADDress:CONDition` is set to `INRange` or `OORange`.

Parameters:

<SrcAddrPattMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:LENGth:CONDition <SearchName>,<TypeOperator>
SEARch:TRIGger:HBTO:LENGth:CONDition? <SearchName>

Sets the condition for the length / type. You can define an exact value or a value range.

Parameters:

<TypeOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:HBTO:LENGth:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:HBTO:LENGth:MIN` and `SEARch:TRIGger:HBTO:LENGth:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:LENGth:MIN <SearchName>,<TypePatternMin>
SEARch:TRIGger:HBTO:LENGth:MIN? <SearchName>

Specifies the length / type bit pattern, or sets the start value of a pattern range.

Parameters:

<TypePatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:LENGth:MAX <SearchName>,<TypePatternMax>
SEARch:TRIGger:HBTO:LENGth:MAX? <SearchName>

Sets the end value of the length / type range if `SEARch:TRIGger:HBTO:LENGth:CONDition` is set to `INRange` or `OORange`.

Parameters:

<TypePatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:CRC:CONDition <SearchName>,<CRCOperator>
SEARch:TRIGger:HBTO:CRC:CONDition? <SearchName>

Sets the condition for the frame check. You can define an exact value or a value range.

Parameters:

<CRCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:HBTO:CRC:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:HBTO:CRC:MIN` and `SEARch:TRIGger:HBTO:CRC:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:CRC:MIN <SearchName>,<CRCPatternMin>
SEARch:TRIGger:HBTO:CRC:MIN? <SearchName>

Specifies the frame check bit pattern, or sets the start value of a pattern range.

Parameters:

<CRCPatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:CRC:MAX <SearchName>,<CRCPatternMax>
SEARch:TRIGger:HBTO:CRC:MAX? <SearchName>

Sets the end value of the frame check range if `SEARch:TRIGger:HBTO:CRC:CONDition` is set to `INRange` or `OORange`.

Parameters:

<CRCPatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:DATA:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:HBTO:DATA:DCONdition? <SearchName>

Sets the condition for the data. You can define an exact value or a value range.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:HBTO:DATA:DMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:HBTO:DATA:DMIN` and `SEARch:TRIGger:HBTO:DATA:DMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:DATA:DMIN <SearchName>,<DataMin>
SEARch:TRIGger:HBTO:DATA:DMIN? <SearchName>

Specifies the data bit pattern, or sets the start value of a pattern range.

Parameters:

<DataMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:DATA:DMAX <SearchName>,<DataMax>
SEARch:TRIGger:HBTO:DATA:DMAX? <SearchName>

Sets the end value of the data range if `SEARch:TRIGger:HBTO:DATA:DCONdition` is set to `INRange` or `OORange`.

Parameters:

<DataMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:DATA:ICONdition <SearchName>,<DataIdxOperator>
SEARch:TRIGger:HBTO:DATA:ICONdition? <SearchName>

Sets the condition for the index. You can define an exact value or a value range.

Parameters:

<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:HBTO:DATA:IMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:HBTO:DATA:IMIN` and `SEARch:TRIGger:HBTO:DATA:IMAX`.

*RST: INRange

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:DATA:IMIN <SearchName>,<DataIndexMin>
SEARch:TRIGger:HBTO:DATA:IMIN? <SearchName>

Specifies the index bit pattern, or sets the start value of a pattern range.

Parameters:

<DataIndexMin> Range: 1 to 0
Increment: 1
*RST: 1

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:DATA:IMAX <SearchName>,<DataIndexMax>
SEARch:TRIGger:HBTO:DATA:IMAX? <SearchName>

Sets the end value of the index range if `SEARch:TRIGger:HBTO:DATA:ICONdition` is set to `INRange` or `OORange`.

Parameters:

<DataIndexMax>	Range:	1 to 0
	Increment:	1
	*RST:	0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:ERRor:PREamble <SearchName>,<ErrorPreamble>
SEARch:TRIGger:HBTO:ERRor:PREamble? <SearchName>

Enables / disables search for preamble error.

Parameters:

<ErrorPreamble>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:ERRor:CRC <SearchName>,<ErrorCRC>
SEARch:TRIGger:HBTO:ERRor:CRC? <SearchName>

Enables / disables trigger on Cyclic Redundancy Check (CRC) error.

Parameters:

<ErrorCRC>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:HBTO:ERRor:SFD <SearchName>,<ErrorSFD>
SEARch:TRIGger:HBTO:ERRor:SFD? <SearchName>

Enables / disables search for start frame delimiter (SFD) error.

Parameters:

<ErrorSFD>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

24.17.19.5 Search results

In all `SEARch:RESult:HBTO:FRAMe<m>` commands, the suffix <m> selects the frame number in the list of search results.

SEARch:RESult:HBTO:FCOunt?....2111
SEARch:RESult:HBTO:FRAMe<m>:TYPE?....2111
SEARch:RESult:HBTO:FRAMe<m>:STATe?....2112
SEARch:RESult:HBTO:FRAMe<m>:STARt?....2112
SEARch:RESult:HBTO:FRAMe<m>:STOP?....2112
SEARch:RESult:HBTO:FRAMe<m>:DESTaddress?....2113
SEARch:RESult:HBTO:FRAMe<m>:SRCaddress?....2113
SEARch:RESult:HBTO:FRAMe<m>:DATA?....2113
SEARch:RESult:HBTO:FRAMe<m>:CRC?....2114
SEARch:RESult:HBTO:FRAMe<m>:NUMWords?....2114
SEARch:RESult:HBTO:FRAMe<m>:FTYPe?....2114
SEARch:RESult:HBTO:FRAMe<m>:DSYMbol?....2115
SEARch:RESult:HBTO:FRAMe<m>:SSYMbol?....2115
SEARch:RESult:HBTO:FRAMe<m>:WORD<n>:TYPE?....2115
SEARch:RESult:HBTO:FRAMe<m>:WORD<n>:VALue?....2116

SEARch:RESult:HBTO:FCOunt? <SearchName>

Returns the number of decoded frames within the search result.

Query parameters:
<SearchName>

Return values:

<FrameCount>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:TYPE? <SearchName>

Returns the value of length / type field of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<TypeLen>	Range:	0 to 65535
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:STATe? <SearchName>

Returns the state of the frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> OK | ERR_PREAMBLE | ERR_SFD | ERR_LENGTH | ERR_CRC | UNCorrelated | INComplete

OK
Valid frame.

ERR_PREAMBLE
Erroneous frame due to preamble error.

ERR_SFD
Erroneous frame due to SFD error.

ERR_LENGTH
Erroneous frame due to length / type error.

ERR_CRC
Erroneous frame due to CRC error.

*RST: OK

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:STARt? <SearchName>

Returns the start time of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:STOP? <SearchName>

Returns the end time of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Stop>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:DESTaddress? <SearchName>

Returns the destination address of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<DestAddress>

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:SRCaddress? <SearchName>

Returns the source address of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SourceAddress>

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:DATA? <SearchName>

Returns the data for the selected frame, corresponds to the Data column in the decode results table.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:CRC? <SearchName>

Returns the CRC checksum of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<CRC> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:NUMWords? <SearchName>

Returns the number of decoded words for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<NumWords> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:FTYPe? <SearchName>

Returns the type of frame for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameType> IDLE | MAC | FILLer | UNKNown

IDLE
IDLE frame. This frame is used for clock synchronization.

MAC
MAC frame. This frame contains information that define how to go about transmitting and receiving frames.

FILLer
Filler frame. The frame is used to maintain transmission activity.

UNKNown
No meaningful frame can be determined.

*RST: MAC

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:DSYMbol? <SearchName>

Returns the destination symbols of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<DestTranslation>

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:SSYMbol? <SearchName>

Returns the source symbols of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SrcTranslation>

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:WORD<n>:TYPE? <SearchName>

Returns the data type of the selected word within the search result.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<WordType> String parameter

Usage: Query only

SEARch:RESult:HBTO:FRAMe<m>:WORD<n>:VALue? <SearchName>

Returns the data value of the selected word within the search result.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<WordValue> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

24.17.20 Ethernet 1000BASE-T1 (option R&S RTO6-K560)

- Configuration....2116
- Decode results....2119
- Trigger....2125
- Search settings....2133
- Search results....2141

24.17.20.1 Configuration

In all `BUS<m>:TBTO` commands, the suffix <m> selects the serial bus.

BUS<m>:TBTO:ATTN <Attenuation>

Sets the attenuation factor. It is used to de-amplify the reverse signal before subtracting it from the forward signal.

Suffix:
<m> 1..4

Parameters:
<Attenuation> Range: -100 to 0
Increment: 0.1
*RST: -26
Default unit: dB

BUS<m>:TBTO:FDAM <SourceDAminus>

Selects the DA- source of the provided forward single ended signal, if `BUS<m>:TBTO:FTYP` is set to `SINGle`.

Suffix:
<m> 1..4

Parameters:
<SourceDAminus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

BUS<m>:TBTO:FDAP <SourceDAplus>

Selects the DA+ source of the provided forward single ended signal, if `BUS<m>:TBTO:FTYP` is set to `SINGle`.

Suffix:
<m> 1..4

Parameters:
<SourceDAplus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

BUS<m>:TBTO:FDIF <SrcDiff>

Sets the source of the provided forward differential signal, if `BUS<m>:TBTO:FTYP` is set to `DIFFerential`.

Suffix:
<m> 1..4

Parameters:
<SrcDiff> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

BUS<m>:TBTO:FTYP <SignalType>

Sets the type of forward signal measurement.

Suffix:
<m> 1..4

Parameters:
<SignalType> DIFFerential | SINGle

*RST: DIFFerential

BUS<m>:TBTO:RDAM <SrcRevDAminus>

Selects the DA- source of the provided reversed single ended signal, if `BUS<m>:TBTO:RTYP` is set to `SINGle`.

Suffix:
<m> 1..4

Parameters:
<SrcRevDAminus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

BUS<m>:TBTO:RDAP <SourceRevDAplus>

Selects the DA+ source of the provided reversed single ended signal, if `BUS<m>:TBTO:RTYP` is set to `SINGle`.

Suffix:
<m> 1..4

Parameters:
<SourceRevDAplus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

BUS<m>:TBTO:RDIF <SrcRevDiff>

Sets the source of the provided reversed differential signal, if `BUS<m>:TBTO:RTYP` is set to `DIFFerential`.

Suffix:
<m> 1..4

Parameters:
<SrcRevDiff> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

BUS<m>:TBTO:RTYP <SignalRevType>

Sets the type of reverse signal measurement.

Suffix:
<m> 1..4

Parameters:
<SignalRevType> DISabled | DIFFerential | SINGle

*RST: DISabled

BUS<m>:TBTO:MODE <Mode>

Selects the direction of the full-duplex signal you want to look at.

Suffix:
<m> 1..4

Parameters:
<Mode> MASTer | SLAVe | AUTO

*RST: AUTO

BUS<m>:TBTO:POLarity <Polarity>

Selects the polarity of the data signal.

Suffix:
<m> 1..4

Parameters:

<Polarity> NORMal | INVert

*RST: NORMal

BUS<m>:TBTO:SYMRate <SymbolRate>

Defines the rate of ternary symbols.

Suffix:

<m> 1..4

Parameters:

<SymbolRate> Range: 10 to 1500
Increment: 1E-3
*RST: 750
Default unit: MSymb/s

24.17.20.2 Decode results

To show the results on the screen, use the following commands:

- BUS<m>:RESult on page 1671

BUS<m>:TBTO:RESult:FCOunt?....2119
BUS<m>:TBTO:RESult:FRAMe<n>:BITRate?....2120
BUS<m>:TBTO:RESult:FRAMe<n>:CRC?....2120
BUS<m>:TBTO:RESult:FRAMe<n>:DATA?....2120
BUS<m>:TBTO:RESult:FRAMe<n>:DESTaddress?....2121
BUS<m>:TBTO:RESult:FRAMe<n>:DSYMbol?....2121
BUS<m>:TBTO:RESult:FRAMe<n>:FTYPe?....2121
BUS<m>:TBTO:RESult:FRAMe<n>:NUMWords?....2122
BUS<m>:TBTO:RESult:FRAMe<n>:SRCaddress?....2122
BUS<m>:TBTO:RESult:FRAMe<n>:SSYMbol?....2122
BUS<m>:TBTO:RESult:FRAMe<n>:STARt?....2123
BUS<m>:TBTO:RESult:FRAMe<n>:STATe?....2123
BUS<m>:TBTO:RESult:FRAMe<n>:STOP?....2124
BUS<m>:TBTO:RESult:FRAMe<n>:TYPE?....2124
BUS<m>:TBTO:RESult:FRAMe<n>:WORD<o>:TYPE?....2124
BUS<m>:TBTO:RESult:FRAMe<n>:WORD<o>:VALue?....2125

BUS<m>:TBTO:RESult:FCOunt?

Returns the number of decoded frames.

Suffix:

<m> 1..4
Selects the serial bus.

Return values:

<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:TBTO:RESult:FRAMe<n>:BITRate?

Returns the primary bit rate.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<PrimaryBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

BUS<m>:TBTO:RESult:FRAMe<n>:CRC?

Returns the CRC (frame check) checksum of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<CRC> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:TBTO:RESult:FRAMe<n>:DATA?

Returns the data for the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Data>

Usage: Query only

BUS<m>:TBTO:RESult:FRAMe<n>:DESTaddress?

Returns the destination address of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<DestAddress>

Usage: Query only

BUS<m>:TBTO:RESult:FRAMe<n>:DSYMbol?

Returns the destination symbols of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<DestTranslation>

Usage: Query only

BUS<m>:TBTO:RESult:FRAMe<n>:FTYPe?

Returns the type of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameType> IDLE | MAC | OAM | RSFEC | ZEROTSYM | UNKNown

IDLE
IDLE frame. This frame is used for clock synchronization.

MAC
MAC frame. This frame contains information that define how to go about transmitting and receiving frames.

OAM
Operations, administration, and maintenance frame.

RSFEC
Reed Solomon forward error correction check frame.

ZEROTSYM
Zero terrnary symbol.

UNKNown
No meaningful frame can be determined.

*RST: MAC

Usage: Query only

BUS<m>:TBTO:RESult:FRAMe<n>:NUMWords?

Returns the number of decoded words for the selected frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<NumWords> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:TBTO:RESult:FRAMe<n>:SRCaddress?

Returns the source address of the selected frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<SrcAddress>

Usage: Query only

BUS<m>:TBTO:RESult:FRAMe<n>:SSYMbol?

Returns the source symbols of the selected frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<SrcTranslation>

Usage: Query only

BUS<m>:TBTO:RESult:FRAMe<n>:STARt?

Returns the start time of the selected frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:TBTO:RESult:FRAMe<n>:STATe?

Returns the state of the frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<State> OK | ERRFEC | ERRZERO | ERROOR | ERRCRC | INComplete

OK
Valid frame.

ERRFEC
Reed-Solomon forward error correction.

ERRZERO
Zero ternary symbol pair error.

ERROOR
Values are out of range error.

ERR_CRC
Erroneous frame due to CRC error.

INComplete
Incomplete frame.

*RST: OK

Usage: Query only

BUS<m>:TBTO:RESult:FRAMe<n>:STOP?

Returns the end time of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:TBTO:RESult:FRAMe<n>:TYPE?

Returns the value of length / type field of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<TypeLen> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:TBTO:RESult:FRAMe<n>:WORD<o>:TYPE?

Returns the data type for the selected word.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the word.

Return values:
<WordType>

Usage: Query only

BUS<m>:TBTO:RESult:FRAMe<n>:WORD<o>:VALue?

Returns the data value for the selected word.

Suffix:

<m>	1..4 Selects the serial bus.
<n>	* Selects the frame.
<o>	* Selects the word.

Return values:

<WordValue>	Range:	0 to 65535
	Increment:	1
	*RST:	0

Usage: Query only

24.17.20.3 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:TBTO:TRGLevel 2126
TRIGger<m>:TBTO:TYPE 2126
TRIGger<m>:TBTO:CRC:CONDition 2126
TRIGger<m>:TBTO:CRC:MAX 2127
TRIGger<m>:TBTO:CRC:MIN 2127
TRIGger<m>:TBTO:DADDress:CONDition 2127
TRIGger<m>:TBTO:DADDress:MAX 2128
TRIGger<m>:TBTO:DADDress:MIN 2128
TRIGger<m>:TBTO:DATA:DCONdition 2128
TRIGger<m>:TBTO:DATA:DMAX 2129
TRIGger<m>:TBTO:DATA:DMIN 2129
TRIGger<m>:TBTO:DATA:ICONdition 2129
TRIGger<m>:TBTO:DATA:IMAX 2130
TRIGger<m>:TBTO:DATA:IMIN 2130
TRIGger<m>:TBTO:ERRor:CRC 2130

TRIGger<m>:TBTO:ERRor:FEC....2131
TRIGger<m>:TBTO:ERRor:OOR....2131
TRIGger<m>:TBTO:ERRor:ZERO....2131
TRIGger<m>:TBTO:LENGth:CONDition....2131
TRIGger<m>:TBTO:LENGth:MAX....2132
TRIGger<m>:TBTO:LENGth:MIN....2132
TRIGger<m>:TBTO:SADDress:CONDition....2132
TRIGger<m>:TBTO:SADDress:MAX....2133
TRIGger<m>:TBTO:SADDress:MIN....2133

TRIGger<m>:TBTO:TRGLevel <TriggerLevel>

Sets the trigger level.

Suffix:
<m> 1..3

Parameters:
<TriggerLevel> Range: -0.5 to 0.5
Increment: 1E-3
*RST: 0.01
Default unit: V

TRIGger<m>:TBTO:TYPE <Type>

Selects the type of frame to be triggered on.

Suffix:
<m> 1..3

Parameters:
<Type> STARt | MAC | IDLE | ERRor

STARt
Start of frame.

MAC
MAC frame. This frame contains information that define how to go about transmitting and receiving frames.

IDLE
IDLE frame. This frame is used for clock synchronization.

ERRor
Error frame. You can enable the error to be triggered on:
`TRIGger<m>:TBTO:ERRor:CRC`
`TRIGger<m>:TBTO:ERRor:FEC`
`TRIGger<m>:TBTO:ERRor:OOR`
`TRIGger<m>:TBTO:ERRor:ZERO`

*RST: STARt

TRIGger<m>:TBTO:CRC:CONDition <CRCOperator>

Sets the condition for the frame check. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<CRCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:TBTO:CRC:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:TBTO:CRC:MIN` and `TRIGger<m>:TBTO:CRC:MAX` .

*RST: EQUal

TRIGger<m>:TBTO:CRC:MAX <CRCPatternMax>

Sets the end value of the frame check range if `TRIGger<m>:TBTO:CRC:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<CRCPatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:TBTO:CRC:MIN <CRCPatternMin>

Specifies the frame check bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<CRCPatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:TBTO:DADDress:CONDition <DestAddrOptor>

Sets the condition for the destination address. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DestAddrOptor> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:TBTO:DADDress:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:TBTO:DADDress:MIN` and `TRIGger<m>:TBTO:DADDress:MAX`.

*RST: EQUal

TRIGger<m>:TBTO:DADDress:MAX <DestAddrPattMax>

Sets the end value of the destination address range if `TRIGger<m>:TBTO:DADDress:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<DestAddrPattMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:TBTO:DADDress:MIN <DestAddrPattMin>

Specifies the destination address bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DestAddrPattMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:TBTO:DATA:DCONdition <DataOperator>

Sets the condition for the data. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:TBTO:DATA:DMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:TBTO:DATA:DMIN` and `TRIGger<m>:TBTO:DATA:DMAX`.

*RST: EQUal

TRIGger<m>:TBTO:DATA:DMAX <DataMax>

Sets the end value of the data range if `TRIGger<m>:TBTO:DATA:DCONdition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<DataMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:TBTO:DATA:DMIN <DataMin>

Specifies the data bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DataMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:TBTO:DATA:ICONdition <DataIdxOperator>

Sets the condition for the index. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:TBTO:DATA:IMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:TBTO:DATA:IMIN` and `TRIGger<m>:TBTO:DATA:IMAX`.

*RST: INRange

TRIGger<m>:TBTO:DATA:IMAX <DataIndexMax>

Sets the end value of the index range if `TRIGger<m>:TBTO:DATA:ICONdition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<DataIndexMax> Range: 1 to 0
Increment: 1
*RST: 0

TRIGger<m>:TBTO:DATA:IMIN <DataIndexMin>

Specifies the index bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<DataIndexMin> Range: 1 to 0
Increment: 1
*RST: 1

TRIGger<m>:TBTO:ERRor:CRC <ErrorCRC>

Enables triggering on cyclic redundancy check (CRC) error.

Suffix:
<m> 1..3

Parameters:
<ErrorCRC> ON | OFF
*RST: ON

TRIGger<m>:TBTO:ERRor:FEC <ErrorFEC>

Enables triggering, if more than 22 RS-FEC errors are detected.

Suffix:
<m> 1..3

Parameters:
<ErrorFEC> ON | OFF
*RST: ON

TRIGger<m>:TBTO:ERRor:OOR <ErrorOOR>

Enables triggering if values are out of range (OOR). For example, a preamble error, or invalid symbols in control blocks.

Suffix:
<m> 1..3

Parameters:
<ErrorOOR> ON | OFF
*RST: ON

TRIGger<m>:TBTO:ERRor:ZERO <ErrorZERO>

Enables triggering, if a zero ternary symbol pair (0-0) is detected.

Suffix:
<m> 1..3

Parameters:
<ErrorZERO> ON | OFF
*RST: ON

TRIGger<m>:TBTO:LENGth:CONDition <TypeOperator>

Sets the condition for the length / type. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<TypeOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:TBTO:LENGth:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:TBTO:LENGth:MIN` and `TRIGger<m>:TBTO:LENGth:MAX`.

*RST: EQUal

TRIGger<m>:TBTO:LENGth:MAX <TypePatternMax>

Sets the end value of the length / type range if `TRIGger<m>:TBTO:LENGth:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<TypePatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:TBTO:LENGth:MIN <TypePatternMin>

Specifies the length / type bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<TypePatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:TBTO:SADDress:CONDition <SrcAddrOperator>

Sets the condition for the source address. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<SrcAddrOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:TBTO:SADDress:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:TBTO:SADDress:MIN` and `TRIGger<m>:TBTO:SADDress:MAX`.

*RST: EQUal

TRIGger<m>:TBTO:SADDress:MAX <SrcAddrPattMax>

Sets the end value of the source address range, if `TRIGger<m>:TBTO:SADDress:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<SrcAddrPattMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

TRIGger<m>:TBTO:SADDress:MIN <SrcAddrPattMin>

Specifies the source address bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<SrcAddrPattMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

24.17.20.4 Search settings

SEARch:TRIGger:TBTO:CRC:CONDition....2134
SEARch:TRIGger:TBTO:CRC:MAX....2134
SEARch:TRIGger:TBTO:CRC:MIN....2135
SEARch:TRIGger:TBTO:DADDress:CONDition....2135
SEARch:TRIGger:TBTO:DADDress:MAX....2135
SEARch:TRIGger:TBTO:DADDress:MIN....2136
SEARch:TRIGger:TBTO:DATA:DCONdition....2136
SEARch:TRIGger:TBTO:DATA:DMAX....2136
SEARch:TRIGger:TBTO:DATA:DMIN....2137
SEARch:TRIGger:TBTO:DATA:ICONdition....2137
SEARch:TRIGger:TBTO:DATA:IMAX....2137
SEARch:TRIGger:TBTO:DATA:IMIN....2138

SEARch:TRIGger:TBTO:ERRor:CRC....2138
SEARch:TRIGger:TBTO:ERRor:FEC....2138
SEARch:TRIGger:TBTO:ERRor:OOR....2138
SEARch:TRIGger:TBTO:ERRor:ZERO....2139
SEARch:TRIGger:TBTO:LENGth:CONDition....2139
SEARch:TRIGger:TBTO:LENGth:MAX....2139
SEARch:TRIGger:TBTO:LENGth:MIN....2140
SEARch:TRIGger:TBTO:SADDress:CONDition....2140
SEARch:TRIGger:TBTO:SADDress:MAX....2140
SEARch:TRIGger:TBTO:SADDress:MIN....2141
SEARch:TRIGger:TBTO:TYPE....2141

SEARch:TRIGger:TBTO:CRC:CONDition <SearchName>,<CRCOperator>
SEARch:TRIGger:TBTO:CRC:CONDition? <SearchName>

Sets the condition for the frame check. You can define an exact value or a value range.

Parameters:

<CRCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:TBTO:CRC:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:TBTO:CRC:MIN` and `SEARch:TRIGger:TBTO:CRC:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:CRC:MAX <SearchName>,<CRCPatternMax>
SEARch:TRIGger:TBTO:CRC:MAX? <SearchName>

Sets the end value of the frame check range if `SEARch:TRIGger:TBTO:CRC:CONDition` is set to `INRange` or `OORange`.

Parameters:

<CRCPatternMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:CRC:MIN <SearchName>,<CRCPatternMin>
SEARch:TRIGger:TBTO:CRC:MIN? <SearchName>

Specifies the frame check bit pattern, or sets the start value of a pattern range.

Parameters:

<CRCPatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:DADDress:CONDition <SearchName>,<DestAddrOptor>
SEARch:TRIGger:TBTO:DADDress:CONDition? <SearchName>

Sets the condition for the destination address. You can define an exact value or a value range.

Parameters:

<DestAddrOptor> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:TBTO:DADDress:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:TBTO:DADDress:MIN` and `SEARch:TRIGger:TBTO:DADDress:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:DADDress:MAX <SearchName>,<DestAddrPattMax>
SEARch:TRIGger:TBTO:DADDress:MAX? <SearchName>

Sets the end value of the destination address range if `SEARch:TRIGger:TBTO:DADDress:CONDition` is set to `INRange` or `OORange`.

Parameters:

<DestAddrPattMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:DADDress:MIN <SearchName>,<DestAddrPattMin>
SEARch:TRIGger:TBTO:DADDress:MIN? <SearchName>

Specifies the destination address bit pattern, or sets the start value of a pattern range.

Parameters:

<DestAddrPattMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:DATA:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:TBTO:DATA:DCONdition? <SearchName>

Sets the condition for the data. You can define an exact value or a value range.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:TBTO:DATA:DMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:TBTO:DATA:DMIN` and `SEARch:TRIGger:TBTO:DATA:DMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:DATA:DMAX <SearchName>,<DataMax>
SEARch:TRIGger:TBTO:DATA:DMAX? <SearchName>

Sets the end value of the data range if `SEARch:TRIGger:TBTO:DATA:DCONdition` is set to `INRange` or `OORange`.

Parameters:

<DataMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:DATA:DMIN <SearchName>,<DataMin>
SEARch:TRIGger:TBTO:DATA:DMIN? <SearchName>

Specifies the data bit pattern, or sets the start value of a pattern range.

Parameters:

<DataMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:DATA:ICONdition <SearchName>,<DataIdxOperator>
SEARch:TRIGger:TBTO:DATA:ICONdition? <SearchName>

Sets the search condition for the index. You can define an exact value or a value range.

Parameters:

<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:TBTO:DATA:IMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:TBTO:DATA:IMIN` and `SEARch:TRIGger:TBTO:DATA:IMAX`.

*RST: INRange

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:DATA:IMAX <SearchName>,<DataIndexMax>
SEARch:TRIGger:TBTO:DATA:IMAX? <SearchName>

Sets the end value of the index range to be searched for, if `SEARch:TRIGger:TBTO:DATA:IMAX` is set to `INRange` or `OORange`.

Parameters:

<DataIndexMax> Range: 1 to 0
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:DATA:IMIN <SearchName>,<DataIndexMin>
SEARch:TRIGger:TBTO:DATA:IMIN? <SearchName>

Specifies the index bit pattern, or sets the start value of a pattern range.

Parameters:

<DataIndexMin>	Range:	1 to 0
	Increment:	1
	*RST:	1

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:ERRor:CRC <SearchName>,<ErrorCRC>
SEARch:TRIGger:TBTO:ERRor:CRC? <SearchName>

Enables search for cyclic redundancy check (CRC) error.

Parameters:

<ErrorCRC>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:ERRor:FEC <SearchName>,<ErrorFEC>
SEARch:TRIGger:TBTO:ERRor:FEC? <SearchName>

Enables search for RS-FEC errors.

Parameters:

<ErrorFEC>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:ERRor:OOR <SearchName>,<ErrorOOR>
SEARch:TRIGger:TBTO:ERRor:OOR? <SearchName>

Enables search for values that are out of range.

Parameters:

<ErrorOOR>	ON | OFF	
	*RST:	ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:ERRor:ZERO <SearchName>,<ErrorZERO>
SEARch:TRIGger:TBTO:ERRor:ZERO? <SearchName>

Enables search for zero ternary symbol pairs error.

Parameters:
<ErrorZERO> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:LENGth:CONDition <SearchName>,<TypeOperator>
SEARch:TRIGger:TBTO:LENGth:CONDition? <SearchName>

Sets the condition for the length/ type. You can define an exact value or a value range.

Parameters:
<TypeOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, not equal, less than, less or equal than, greater than, greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:TBTO:LENGth:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:TBTO:LENGth:MIN` and `SEARch:TRIGger:TBTO:LENGth:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:LENGth:MAX <SearchName>,<TypePatternMax>
SEARch:TRIGger:TBTO:LENGth:MAX? <SearchName>

Sets the end value of the length / type range if `SEARch:TRIGger:TBTO:LENGth:CONDition` is set to `INRange` or `OORange`.

Parameters:
<TypePatternMax>

Parameters for setting and query:
<SearchName> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

SEARch:TRIGger:TBTO:LENGth:MIN <SearchName>,<TypePatternMin>
SEARch:TRIGger:TBTO:LENGth:MIN? <SearchName>

Specifies the length/ type bit pattern, or sets the start value of a pattern range.

Parameters:

<TypePatternMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:SADDress:CONDition <SearchName>,<SrcAddrOperator>
SEARch:TRIGger:TBTO:SADDress:CONDition? <SearchName>

Sets the condition for the source address. You can define an exact value or a value range.

Parameters:

<SrcAddrOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, not equal, less than, less or equal than, greater than, greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:TBTO:SADDress:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:TBTO:SADDress:MIN` and`SEARch:TRIGger:TBTO:SADDress:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:SADDress:MAX <SearchName>,<SrcAddrPattMax>
SEARch:TRIGger:TBTO:SADDress:MAX? <SearchName>

Sets the end value of the source address range if `SEARch:TRIGger:TBTO:SADDress:CONDition` is set to `INRange` or `OORange`.

Parameters:

<SrcAddrPattMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:SADDress:MIN <SearchName>,<SrcAddrPattMin>
SEARch:TRIGger:TBTO:SADDress:MIN? <SearchName>

Specifies the source address bit pattern, or sets the start value of a pattern range.

Parameters:

<SrcAddrPattMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (do not care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:TBTO:TYPE <SearchName>,<Type>
SEARch:TRIGger:TBTO:TYPE? <SearchName>

Selects the type of frame to be searched for.

Parameters:

<Type> STARt | MAC | IDLE | ERRor

STARt
Start of frame.

MAC
MAC frame. This frame contains information that define how to go about transmitting and receiving frames.

IDLE
IDLE frame. This frame is used for clock synchronization.

ERRor
Error frame. This frame contains erroneous bits.

*RST: STARt

Parameters for setting and query:
<SearchName>

24.17.20.5 Search results

SEARch:RESult:TBTO:FCOunt?....2142
SEARch:RESult:TBTO:FRAMe<m>:CRC?....2142
SEARch:RESult:TBTO:FRAMe<m>:DATA?....2142
SEARch:RESult:TBTO:FRAMe<m>:DESTaddress?....2142
SEARch:RESult:TBTO:FRAMe<m>:DSYMbol?....2143
SEARch:RESult:TBTO:FRAMe<m>:FTYPe?....2143
SEARch:RESult:TBTO:FRAMe<m>:NUMWords?....2144
SEARch:RESult:TBTO:FRAMe<m>:SRCaddress?....2144
SEARch:RESult:TBTO:FRAMe<m>:SSYMbol?....2144
SEARch:RESult:TBTO:FRAMe<m>:STARt?....2145
SEARch:RESult:TBTO:FRAMe<m>:STATe?....2145
SEARch:RESult:TBTO:FRAMe<m>:STOP?....2145
SEARch:RESult:TBTO:FRAMe<m>:TYPE?....2146

SEARch:RESult:TBTO:FRAMe<m>:WORD<n>:TYPE?..... 2146
SEARch:RESult:TBTO:FRAMe<m>:WORD<n>:VALue?..... 2146
SEARch:RESult:TBTO:FRAMe<m>:WORD<n>:VSTR?..... 2147

SEARch:RESult:TBTO:FCOunt? <SearchName>

Returns the number of decoded frames within the search result.

Query parameters:
<SearchName>

Return values:

<FrameCount>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:TBTO:FRAMe<m>:CRC? <SearchName>

Returns the CRC checksum of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<CRC>	Range:	0 to 4294967295
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:TBTO:FRAMe<m>:DATA? <SearchName>

Returns the data for the selected frame.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:TBTO:FRAMe<m>:DESTaddress? <SearchName>

Returns the destination address of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<DestAddrstr>

Usage: Query only

SEARch:RESult:TBTO:FRAMe<m>:DSYMbol? <SearchName>

Returns the destination symbols of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<DestTranslation>

Usage: Query only

SEARch:RESult:TBTO:FRAMe<m>:FTYPe? <SearchName>

Returns the type of frame for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameType> IDLE | MAC | OAM | RSFEC | BH | CTLADDR | CTLCODE | ZEROTSYM | UNKNown

IDLE
IDLE frame. This frame is used for clock synchronization.

MAC
MAC frame. This frame contains information that define how to go about transmitting and receiving frames.

OAM
Operations, administration and maintenance (OAM) frame.

RSFEC
Reed Solomon forward error correction check frame.

BH
BASE-H fame.

CTLADDR
Control address frame.

CTLCODE
Control code frame.

ZEROTSYM
Zero terrnary symbol frame.

UNKNown
No meaningful frame can be determined.

*RST: MAC

Usage: Query only

SEARch:RESult:TBTO:FRAMe<m>:NUMWords? <SearchName>

Returns the number of decoded words for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<NumWords> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:TBTO:FRAMe<m>:SRCaddress? <SearchName>

Returns the source address of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SrcAddrstr>

Usage: Query only

SEARch:RESult:TBTO:FRAMe<m>:SSYMbol? <SearchName>

Returns the source symbols of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SrcTranslation>

Usage: Query only

SEARch:RESult:TBTO:FRAMe<m>:STARt? <SearchName>

Returns the start time of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Start>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:TBTO:FRAMe<m>:STATe? <SearchName>

Returns the state of the frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> OK | ERRFEC | ERRZERO | ERROOR | ERRCRC | INComplete

OK
Valid frame.

ERRFEC
Reed-Solomon forward error correction.

ERRZERO
Zero ternary symbol pair error.

ERROOR
Values are out of range error.

ERR_CRC
Erroneous frame due to CRC error.

INComplete
Incomplete frame.

*RST: OK

Usage: Query only

SEARch:RESult:TBTO:FRAMe<m>:STOP? <SearchName>

Returns the end time of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:TBTO:FRAMe<m>:TYPE? <SearchName>

Returns the value of length / type field of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<TypeLen> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:TBTO:FRAMe<m>:WORD<n>:TYPE? <SearchName>

Returns the data type of the selected word within the search result.

Suffix:
<m> *
Selects the frame.

<n> *
Selects the word.

Query parameters:
<SearchName>

Return values:
<WordType>

Usage: Query only

SEARch:RESult:TBTO:FRAMe<m>:WORD<n>:VALue? <SearchName>

Returns the data value of the selected word within the search result.

Suffix:
<m> *
Selects the frame.

<n> *
Selects the word.

Query parameters:
<SearchName>

Return values:
<WordValue> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:TBTO:FRAMe<m>:WORD<n>:VSTR? <SearchName>

Returns the string equivalent of data value for the selected cell.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<WordValueString>

Usage: Query only

24.17.21 USB (option R&S RTO6-K570)

- Configuration....2147
- Trigger....2151
- Decode results....2162
- Search settings....2168
- Search results....2183

24.17.21.1 Configuration

In all `BUS<m>:USB` commands, the suffix <m> selects the serial bus.

BUS<m>:USB:TECHnology....2148
BUS<m>:USB:DPLus:SOURce....2148
BUS<m>:USB:DMINus:SOURce....2148
BUS<m>:USB:DIFFerential:SOURce....2149
BUS<m>:USB:DATA:SOURce....2149
BUS<m>:USB:STRobe:SOURce....2149
BUS<m>:USB:DPLus:THReshold....2149

BUS<m>:USB:DMINus:THReshold....2150
BUS<m>:USB:DIFFerential:THReshold....2150
BUS<m>:USB:DATA:THReshold.... 2150
BUS<m>:USB:STRobe:THReshold....2150

BUS<m>:USB:TECHnology <ProtocolType>

Defines the USB protocol technology and transmission speed.

Suffix:
<m> 1..4

Parameters:
<ProtocolType> LOW | FULL | HIGH | | HSIC

LOW
USB low speed protocol (1.5 Mbit/s)

FULL
USB full speed protocol (12 Mbit/s)

HIGH
USB high speed protocol (480 Mbit/s)

HSIC
USB high speed inter-chip (HSIC) protocol (480 Mbit/s)

*RST: LOW

BUS<m>:USB:DPLus:SOURce <SourceDplus>

Selects the source for the D+ data signal (in USB low speed and USB full speed protocol, only).

Suffix:
<m> 1..4

Parameters:
<SourceDplus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: C1W1

Usage: Asynchronous command

BUS<m>:USB:DMINus:SOURce <SourceDminus>

Selects the source for the D- data signal (in USB low speed and USB full speed protocol, only).

Suffix:
<m> 1..4

Parameters:
<SourceDminus> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: C2W1

Usage: Asynchronous command

BUS<m>:USB:DIFFerential:SOURce <SrcDiff>

Selects the source for the differential signal in the USB high speed protocol.

Suffix:
<m> 1..4

Parameters:
<SrcDiff> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: C1W1

Usage: Asynchronous command

BUS<m>:USB:DATA:SOURce <SourceData>

Selects the source for the data signal in the USB HSIC protocol.

Suffix:
<m> 1..4

Parameters:
<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: C1W1

Usage: Asynchronous command

BUS<m>:USB:STRobe:SOURce <SourceStrobe>

Selects the source for the strobe signal in the USB HSIC protocol.

Suffix:
<m> 1..4

Parameters:
<SourceStrobe> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: C2W1

Usage: Asynchronous command

BUS<m>:USB:DPLus:THReshold <ThresholdDplus>

Defines the threshold level for the D+ data signal (in USB low speed and USB full speed protocol, only).

Suffix:
<m> 1..4

Parameters:

<ThresholdDplus>	Range:	-5 to 5
	Increment:	0.01
	*RST:	1.55
	Default unit:	V

BUS<m>:USB:DMINus:THReshold <ThresholdDminus>

Defines the threshold level for the D- data signal (in USB low speed and USB full speed protocol, only).

Suffix:

<m> 1..4

Parameters:

<ThresholdDminus>	Range:	-5 to 5
	Increment:	0.01
	*RST:	1.55
	Default unit:	V

BUS<m>:USB:DIFFerential:THReshold <ThresDiff>

Defines the threshold level for the differential signal in the USB high speed protocol.

Suffix:

<m> 1..4

Parameters:

<ThresDiff>	Range:	-2 to 2
	Increment:	0.01
	*RST:	0
	Default unit:	V

BUS<m>:USB:DATA:THReshold <ThresholdData>

Defines the threshold level for the data signal in the USB HSIC protocol.

Suffix:

<m> 1..4

Parameters:

<ThresholdData>	Range:	-2 to 2
	Increment:	0.01
	*RST:	0.65
	Default unit:	V

BUS<m>:USB:STRobe:THReshold <ThresholdStrobe>

Defines the threshold level for the strobe signal in the USB HSIC protocol.

Suffix:

<m> 1..4

Parameters:

<ThresholdStrobe>	Range:	-2 to 2
	Increment:	0.01
	*RST:	0.65
	Default unit:	V

24.17.21.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:USB:TYPE 2152
TRIGger<m>:USB:ACONdition 2153
TRIGger<m>:USB:AMIN 2153
TRIGger<m>:USB:AMAX 2153
TRIGger<m>:USB:DATA 2154
TRIGger<m>:USB:DCONdition 2154
TRIGger<m>:USB:DPOPerator 2154
TRIGger<m>:USB:DPOSition 2154
TRIGger<m>:USB:ECONdition 2155
TRIGger<m>:USB:EMIN 2155
TRIGger<m>:USB:EMAX 2155
TRIGger<m>:USB:ERRC 2155
TRIGger<m>:USB:FCONdition 2156
TRIGger<m>:USB:FMIN 2156
TRIGger<m>:USB:FMAX 2157
TRIGger<m>:USB:HAND 2157
TRIGger<m>:USB:PATT 2157
TRIGger<m>:USB:PCONdition 2157
TRIGger<m>:USB:PMIN 2158
TRIGger<m>:USB:PMAX 2158
TRIGger<m>:USB:SCONdition 2158
TRIGger<m>:USB:SMIN 2158
TRIGger<m>:USB:SMAX 2159
TRIGger<m>:USB:SPEC 2159
TRIGger<m>:USB:STCO 2159
TRIGger<m>:USB:TCONdition 2160
TRIGger<m>:USB:TMIN 2160
TRIGger<m>:USB:TMAX 2160
TRIGger<m>:USB:TOKen 2160
TRIGger<m>:USB:WADD 2161
TRIGger<m>:USB:WEND 2161
TRIGger<m>:USB:WETCheck 2161

TRIGger<m>:USB:WFRN 2161
TRIGger<m>:USB:WPAY 2161
TRIGger<m>:USB:WPID 2162
TRIGger<m>:USB:WPOR 2162
TRIGger<m>:USB:WSEU 2162
TRIGger<m>:USB:WSTC 2162

TRIGger<m>:USB:TYPE <Type>

Selects the trigger type for USB analysis. The available trigger types depend on the activated USB protocol type.

Parameters:

<Type> SOP | EOP | RST | SUSPend | RESume | TOKen | DATA | HANDshake | SPECial | ERRCond

SOP
Sets the trigger to the SOP (start of packet) field. The start of packet condition is the end of the SYNC field. The trigger instant is the end of the SOP field.

EOP
Sets the trigger to the EOP (end of packet) field. Not available for USB High Speed and USB HSIC protocol types. The trigger instant is the beginning of the EOP field.

RST
Sets the trigger to the Reset field. Not available for USB High Speed and USB HSIC protocol types. For more information on the reset condition, see the USB standard. The trigger instant is the end of the 10 ms period after the SE0 field.

SUSPend
Sets the trigger to the Suspend field. Not available for USB High Speed and USB HSIC protocol types. For more information on the suspend condition, see the USB standard. The trigger instant will be declared after the defined 3 ms timeout.

RESume
Sets the trigger to the Resume field. Not available for USB High Speed and USB HSIC protocol types. For more information on the resume condition, see the USB standard. The trigger instant will be declared after the defined 20 ms timeout.

TOKen
Sets the trigger to one out of four different token trigger types: OUT, IN, SOF, or SETUP.

DATA
Sets the trigger to one out of four different data trigger types: DATA0, DATA1, DATA2, or MDATA.

HANDshake
Sets the trigger to one out of four different handshake trigger types: ACK, NAK, STALL, or NYET.

SPECial
Sets the trigger to one out of four different Special PID trigger types: PREamble, ERR, SPLIT, or PING.

ERRCond
Sets the trigger to one out of seven different error condition trigger types: PID error, CRC5 error, CRC16 error, Bitstuffing error, Unexpected PID error, SE1 error, or Glitching error.

*RST: SOP

TRIGger<m>:USB:ACONdition <OperatorAddress>

Sets the operator to set a specific address or an address range. The address values are set with `TRIGger<m>:USB:AMIN` and `TRIGger<m>:USB:AMAX`.

Parameters:
<OperatorAddress> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one address value to be set using `TRIGger<m>:USB:AMIN`.

INRange | OORange
In range, out of range. These conditions require an address range to be set using `TRIGger<m>:USB:AMIN` and `TRIGger<m>:USB:AMAX`.

*RST: OFF

TRIGger<m>:USB:AMIN <Address>

Specifies the address, or sets the the start value of an address range. The string parameter does not accept the bit value X (don't care).

Parameters:
<Address>

TRIGger<m>:USB:AMAX <AddressTo>

Sets the the end value of an address range if `TRIGger<m>:USB:ACONdition` is set to `INRange` or `OORange`. The string parameter does not accept the bit value X (don't care).

Parameters:
<AddressTo>

TRIGger<m>:USB:DATA <USBDataType>

Sets the trigger to one out of four different payload data types: DATA0, DATA1, DATA2, or MDATA.

Parameters:

<USBDataType> D0 | D1 | D2 | MD

D0
Sets the trigger to the DATA0 field (even PID).

D1
Sets the trigger to the DATA1 field (odd PID).

D2
Sets the trigger to the DATA2 field (data packet PID for high-speed, high bandwidth isochronous transaction in a microframe).

MD
Sets the trigger to the MDATA field (high-speed data packet PID for split and high bandwidth isochronous transactions).

*RST: D0

TRIGger<m>:USB:DCONdition <OperatorData>

Sets the operator (equal or unequal) to set a specific payload data pattern.

Parameters:

<OperatorData> OFF | ANY | EQUal | NEQual

*RST: EQUal

TRIGger<m>:USB:DPOPerator <DataPosOperator>

Sets the operator (any or equal) for the payload data index position.

Parameters:

<DataPosOperator> ANY | OFF | EQUal

ANY = OFF
The position of the bit pattern within the payload data is not relevant.

EQUal
Sets the operator for specifying a special start position for the bit pattern within the payload data.

*RST: ANY

TRIGger<m>:USB:DPOSition <DataPosition>

Specifies the position within a payload data packet, in which a special data pattern is to be searched.

Parameters:

<DataPosition> Range: 0 to 1024
Increment: 1
*RST: 0

TRIGger<m>:USB:ECONdition <EndpointOptor>

Sets the operator to set a specific endpoint or an endpoint range. The endpoint values are set with `TRIGger<m>:USB:EMIN` and `TRIGger<m>:USB:EMAX`.

Parameters:

<EndpointOptor> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one endpoint value to be set using `TRIGger<m>:USB:EMIN`.

INRange | OORange
In range, out of range. These conditions require a range of endpoint values to be set using `TRIGger<m>:USB:EMIN` and `TRIGger<m>:USB:EMAX`.

*RST: OFF

TRIGger<m>:USB:EMIN <Endp>

Specifies the endpoint, or sets the the start value of an endpoint range.

Parameters:

<Endp>

TRIGger<m>:USB:EMAX <EndpTo>

Sets the the end value of an endpoint range if `TRIGger<m>:USB:ECONdition` is set to `INRange` or `OORange`.

Parameters:

<EndpTo>

TRIGger<m>:USB:ERRC <ErrorType>

Sets the trigger to an error condition type.

Parameters:

<ErrorType> ANY | PIDerror | CRC5error | CRC16error | BTST | UNEXpid | SE1error | GLITcherr

ANY
Triggers on any of the errors listed below.

PIDerror
Triggers on any packet identifier error.

CRC5error
Triggers on any CRC5 error event.

CRC16error
Triggers on any CRC16 error event.

BTST
Triggers on any bitstuffing error event (erroneous or missing bit stuffing sequence, see USB standard).

UNEXpid
Triggers on any unexpected PID error (illegal PID, that is not allowed in USB low speed and USB full speed protocols, especially PID's announcing packets such as SPLIT, DATA2, MDATA, or other noncompliant packets).

SE1error
Triggers on the illegal bus state Single Ended 1 (SE1 = both lines high).

GLITcherr
Triggers on any glitching error (illegal bit period, see USB standard for the definition of glitching).

*RST: ANY

TRIGger<m>:USB:FCONdition <FrameNoOperator>

Sets the operator to set a specific frame number or a frame number range. The frame number values are set with `TRIGger<m>:USB:FMIN` and `TRIGger<m>:USB:FMAX`.

Parameters:
<FrameNoOperator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one frame number value to be set using `TRIGger<m>:USB:FMIN`.

INRange | OORange
In range, out of range. These conditions require a range of frame number values to be set using `TRIGger<m>:USB:FMIN` and `TRIGger<m>:USB:FMAX`.

*RST: OFF

TRIGger<m>:USB:FMIN <FrameNumber>

Specifies the frame number, or sets the the start value of a frame number range.

Parameters:
<FrameNumber>

TRIGger<m>:USB:FMAX <FrameNumberTo>

Sets the the end value of a frame number range if `TRIGger<m>:USB:FCONdition` is set to `INRange` or `OORange`.

Parameters:
<FrameNumberTo>

TRIGger<m>:USB:HAND <HandshakeType>

Sets the trigger to one out of four different handshake types: ACK, NAK, STALI, or NYET.

Parameters:
<HandshakeType> ACK | NAK | STALI | NYET

ACK
Sets the trigger to the ACK field (acknowledgment of error-free data packet).

NAK
Sets the trigger to the NAK field (non-acknowledgment, no successful data transmission).

STALI
Sets the trigger to the STALL field (endpoint is halted or a control pipe request is not supported).

NYET
Sets the trigger to the NYET field (no response yet from receiver).

*RST: ACK

TRIGger<m>:USB:PATT <PayloadMuster>

Specifies the payload data pattern that is to be searched.

Parameters:
<PayloadMuster>

TRIGger<m>:USB:PCONdition <OperatorPort>

Sets the operator to set a specific port number or a port number range. The port number values are set with `TRIGger<m>:USB:PMIN` and `TRIGger<m>:USB:PMAX`.

Parameters:
<OperatorPort> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one port number to be set using `TRIGger<m>:USB:PMIN`.

INRange | OORange
In range, out of range. These conditions require a port number range to be set using `TRIGger<m>:USB:PMIN` and `TRIGger<m>:USB:PMAX`.

*RST: OFF

TRIGger<m>:USB:PMIN <Port>

Specifies the port number, or sets the the start value of a port number range.

Parameters:
<Port>

TRIGger<m>:USB:PMAX <PortTo>

Sets the the end value of a port number range if `TRIGger<m>:USB:PCONdition` is set to `INRange` or `OORange`.

Parameters:
<PortTo>

TRIGger<m>:USB:SCONdition <OperatorSEU>

Sets the operator to set a specific SEU or an SEU range. The SEU values are set with `TRIGger<m>:USB:SMIN` and `TRIGger<m>:USB:SMAX`.

For SEU, see "SEU" on page 831.

Parameters:
<OperatorSEU> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one SEU value to be set using `TRIGger<m>:USB:SMIN`.

INRange | OORange
In range, out of range. These conditions require a range of SEU values to be set using `TRIGger<m>:USB:SMIN` and `TRIGger<m>:USB:SMAX`.

*RST: OFF

TRIGger<m>:USB:SMIN <SEU>

Specifies the SEU, or sets the the start value of an SEU range.

Parameters:
<SEU>

TRIGger<m>:USB:SMAX <SEUTo>

Sets the the end value of an SEU range if `TRIGger<m>:USB:SCONdition` is set to `INRange` or `OORange`.

Parameters:
<SEUTo>

TRIGger<m>:USB:SPEC <USBSpecialType>

Sets the trigger to one out of four different Special PID types: PREamble, ERR, SPLit, or PING.

Parameters:
<USBSpecialType> PREamble | ERR | SPLit | PING

PREamble
Sets the trigger to the PREamble PID

ERR
Sets the trigger to the ERRor PID

SPLit
Sets the trigger to the SPLIT PID (in USB high speed transactions)

PING
Sets the trigger to the PING PID (in USB high speed transactions, flow control probe for a bulk/control endpoint)

*RST: PREamble

TRIGger<m>:USB:STCO <SC>

Sets the trigger to a specific start-split or complete-split transaction endpoint.

Parameters:
<SC> ONE | ZERO | DC

ONE
SC = 1 represents a complete-split (CSPLIT) transaction.

ZERO
SC = 0 represents a start-split (SSPLIT) transaction.

DC
SC = X represents "don't care" (DC)

*RST: DC

TRIGger<m>:USB:TCONdition <OperatorET>

Sets the operator to set a specific endpoint type (ET) or an ET range. The ET values are set with `TRIGger<m>:USB:TMIN` and `TRIGger<m>:USB:TMAX`.

Parameters:

<OperatorET> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The position of the endpoint type is not relevant.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one ET value to be set using `TRIGger<m>:USB:TMIN`.

INRange | OORange
In range, out of range. These conditions require a range of ET values to be set using `TRIGger<m>:USB:SMIN` and `TRIGger<m>:USB:SMAX`.

*RST: OFF

TRIGger<m>:USB:TMIN <ET>

Specifies the endpoint type, or sets the the start value of an endpoint type range.

Parameters:

<ET>

TRIGger<m>:USB:TMAX <ETTo>

Sets the the end value of an endpoint type range if `TRIGger<m>:USB:TCONdition` is set to `INRange` or `OORange`.

Parameters:

<ETTo>

TRIGger<m>:USB:TOKen <USBTokenType>

Sets the trigger to one out of four different token types: OUT, IN, SOF, or SETup.

Parameters:

<USBTokenType> OUT | IN | SOF | SETup

OUT
Sets the trigger to the OUT token (OUT packet from host to device).

IN
Sets the trigger to the IN token (IN packet from device to host).

SOF
Sets the trigger to the SOF token (start of frame marker and frame number).

SETup
Sets the trigger to the SETUP token (address and endpoint number in OUT transaction for setup to a control pipe).

*RST: OUT

TRIGger<m>:USB:WADD <AddressCheck>

Defines, whether the address check shall be activated or not.

Parameters:
<AddressCheck> ON | OFF

*RST: OFF

TRIGger<m>:USB:WEND <EndpointCheck>

Defines, whether the endpoint check shall be activated or not.

Parameters:
<EndpointCheck> ON | OFF

*RST: OFF

TRIGger<m>:USB:WETCheck <WithETCheck>

Defines, whether the Endpoint Type (ET) check shall be activated or not.

Parameters:
<WithETCheck> ON | OFF

*RST: OFF

TRIGger<m>:USB:WFRN <FrameNoChk>

Defines, whether the frame number check shall be activated or not.

Parameters:
<FrameNoChk> ON | OFF

*RST: OFF

TRIGger<m>:USB:WPAY <PayloadCheck>

Defines, whether the payload data check shall be activated or not.

Parameters:
<PayloadCheck> ON | OFF

*RST: OFF

TRIGger<m>:USB:WPID <WithPIDCheck>

Defines, whether the packet ID error check shall be activated or not.

Parameters:
<WithPIDCheck> ON | OFF
*RST: OFF

TRIGger<m>:USB:WPOR <WithPortCheck>

Defines, whether the port check shall be activated or not.

Parameters:
<WithPortCheck> ON | OFF
*RST: OFF

TRIGger<m>:USB:WSEU <WithSEUCheck>

Defines, whether the SEU check shall be activated or not.

For SEU, see "SEU" on page 831.

Parameters:
<WithSEUCheck> ON | OFF
*RST: OFF

TRIGger<m>:USB:WSTC <WithSCCheck>

Defines, whether the Start / Complete (SC) check shall be activated or not.

Parameters:
<WithSCCheck> ON | OFF
*RST: OFF

24.17.21.3 Decode results

In all `BUS<m>:USB:PACKet<n>` commands, the suffix <m> selects the serial bus and the suffix <n> selects the packet number in the decode table.

► Query the status of packet #19: `BUS:USB:PACKet19:STAT?`

The result of this remote command query should be "OK".

BUS<m>:USB:PACKet<n>:PID?....2163
BUS<m>:USB:PACKet<n>:ADDRess?....2164
BUS<m>:USB:PACKet<n>:CRC?....2164
BUS<m>:USB:PACKet<n>:DATA?....2165
BUS<m>:USB:PACKet<n>:ENDPoint?....2165
BUS<m>:USB:PACKet<n>:ET?....2165
BUS<m>:USB:PACKet<n>:FRAMe?....2165

BUS<m>:USB:PACKet<n>:PORT?.....2166
BUS<m>:USB:PACKet<n>:SC?.....2166
BUS<m>:USB:PACKet<n>:SEU?.....2166
BUS<m>:USB:PACKet<n>:STARt?.....2167
BUS<m>:USB:PACKet<n>:STATus?.....2167
BUS<m>:USB:PACKet<n>:STOP?.....2167
BUS<m>:USB:PCOunt?.....2168

BUS<m>:USB:PACKet<n>:PID?

Returns the packet PID for the selected serial bus and packet number.

Suffix:

<m> 1..4

<n> *

Return values:

<PID> RES | OUT | ACK | DATA0 | DATa0 | PING | SOF | NYET | DATA2 | DATa2 | SPLIT | SPLit | IN | NAK | DATA1 | DATa1 | PRE | SETUP | SETup | STALL | STALI | MDATA | MDATa | UNK

RES
RES = Reserved

OUT
OUT Token PID

IN
IN Token PID

SOF
Start Of Frame PID

SETUP = SETup
SETUP PID

DATA0 = DATa0
DATA0 PID, even PID

DATA1 = DATa1
DATA1 PID, odd PID

DATA2 = DATa2
DATA2 PID (only valid in USB high speed and USB HSIC protocols)

MDATA = MDATa
MDATA PID (only valid in USB high speed and USB HSIC protocols)

ACK
ACKnowledgment PID

NAK
Non-AcKnowledgment PID

STALL = STALI
STALL PID

NYET
Not ready YET (only valid in USB high speed and USB HSIC protocols)

PRE
PREamble PID (only valid in USB high speed and USB HSIC protocols)

SPLIT = SPLit
SPLIT PID (only valid in USB high speed and USB HSIC protocols)

PING
PING PID (only valid in USB high speed and USB HSIC protocols)

UNK
UNK = Unknown PID

*RST: RES

Usage: Query only

BUS<m>:USB:PACKet<n>:ADDRess?

Returns the packet address for the selected serial bus and packet number.

Suffix:
<m> 1..4

<n> *

Return values:
<Addr> Range: 0 to 127
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:USB:PACKet<n>:CRC?

Returns the packet CRC (Cyclic Redundancy Code) for the selected serial bus and packet number.

Suffix:
<m> 1..4

<n> *

Return values:
<CRC> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:USB:PACKet<n>:DATA?

Returns the payload data from the packet with the selected packet number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

Parameters:

<Data> Comma-separated sequence of integer values (N, D1, D2,…, DN). N is the number of bytes in the packet. and D1…DN are the values of the bytes.

Example:

```
BUS:USB:PACKet4:DATA?
<-- 6,18,52,86,120,154,188
```

Usage: Query only

BUS<m>:USB:PACKet<n>:ENDPoint?

Returns the endpoint for the selected serial bus and packet number.

Suffix:

<m> 1..4

<n> *

Return values:

<Endp>
Range: 0 to 15
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:USB:PACKet<n>:ET?

Returns the endpoint type (ET) for the selected serial bus and packet number.

Suffix:

<m> 1..4

<n> *

Return values:

<ET>
Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:USB:PACKet<n>:FRAMe?

Returns the frame number for the selected serial bus and packet number.

Suffix:
<m> 1..4

<n> *

Return values:
<FrameNo> Range: 0 to 2047
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:USB:PACKet<n>:PORT?

Returns the port number for the selected serial bus and packet number.

Suffix:
<m> 1..4

<n> *

Return values:
<Port> Range: 0 to 127
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:USB:PACKet<n>:SC?

Returns the Start- / Complete-split transaction (SSPLIT / CSPLIT) flag bits for the selected serial bus and packet number.

Suffix:
<m> 1..4

<n> *

Return values:
<SC>

Usage: Query only

BUS<m>:USB:PACKet<n>:SEU?

Returns the SEU values for the selected serial bus and packet number.

For SEU, see "SEU" on page 831.

Suffix:
<m> 1..4

<n> *

Return values:

<SEU> Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:USB:PACKet<n>:STARt?

Returns the start position of the packet with the selected packet number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:USB:PACKet<n>:STATus?

Returns the status of the packet with the selected packet number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

Return values:

<State> OK | PID | CRC | BTST | GLITCH | GLITch | BYTE
*RST: OK

Usage: Query only

BUS<m>:USB:PACKet<n>:STOP?

Returns the stop time of the packet with the selected packet number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

Return values:

<Stop>

Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:USB:PCOunt?

Returns the packet count for the selected serial bus, i.e. the number of packets in the present acquisition.

Suffix:

<m> 1..4

Return values:

<Count>

Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

24.17.21.4 Search settings

SEARch:TRIGger:USB:ACONdition....2169
SEARch:TRIGger:USB:AMIN....2169
SEARch:TRIGger:USB:AMAX....2170
SEARch:TRIGger:USB:BITSterror....2170
SEARch:TRIGger:USB:CRC16error....2170
SEARch:TRIGger:USB:CRC5error....2170
SEARch:TRIGger:USB:DATA....2171
SEARch:TRIGger:USB:DCONdition....2171
SEARch:TRIGger:USB:DPOPerator....2171
SEARch:TRIGger:USB:DPOSition....2172
SEARch:TRIGger:USB:ECONdition....2172
SEARch:TRIGger:USB:EMIN....2173
SEARch:TRIGger:USB:EMAX....2173
SEARch:TRIGger:USB:FCONdition....2173
SEARch:TRIGger:USB:FMIN....2173
SEARch:TRIGger:USB:FMAX....2174
SEARch:TRIGger:USB:GLITcherror....2174
SEARch:TRIGger:USB:HAND....2174
SEARch:TRIGger:USB:PATT....2174
SEARch:TRIGger:USB:PCONdition....2175
SEARch:TRIGger:USB:PMIN....2175
SEARch:TRIGger:USB:PMAX....2175
SEARch:TRIGger:USB:PIDerror....2175
SEARch:TRIGger:USB:SCONdition....2176
SEARch:TRIGger:USB:SMIN....2176
SEARch:TRIGger:USB:SMAX....2176

SEARch:TRIGger:USB:SDATa 2177
SEARch:TRIGger:USB:SERRor 2177
SEARch:TRIGger:USB:SHANdshake 2177
SEARch:TRIGger:USB:SSOP 2177
SEARch:TRIGger:USB:SSPE 2178
SEARch:TRIGger:USB:SPEC 2178
SEARch:TRIGger:USB:STCO 2178
SEARch:TRIGger:USB:STOKen 2179
SEARch:TRIGger:USB:TCONdition 2179
SEARch:TRIGger:USB:TMIN 2179
SEARch:TRIGger:USB:TMAX 2180
SEARch:TRIGger:USB:TOKen 2180
SEARch:TRIGger:USB:WADD 2180
SEARch:TRIGger:USB:WEND 2181
SEARch:TRIGger:USB:WETCheck 2181
SEARch:TRIGger:USB:WFRN 2181
SEARch:TRIGger:USB:WPAY 2181
SEARch:TRIGger:USB:WPID 2182
SEARch:TRIGger:USB:WPOR 2182
SEARch:TRIGger:USB:WSEU 2182
SEARch:TRIGger:USB:WSTC 2182

SEARch:TRIGger:USB:ACONdition <SearchName>,<OperatorAddress>
SEARch:TRIGger:USB:ACONdition? <SearchName>

Sets the operator to set a specific address or an address range.

Parameters:

<OperatorAddress> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one address value to be set using `SEARch:TRIGger:USB:AMIN`.

INRange | OORange
In range, out of range. These conditions require an address range to be set using `SEARch:TRIGger:USB:AMIN` and `SEARch:TRIGger:USB:AMAX`.

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:AMIN <SearchName>,<Address>
SEARch:TRIGger:USB:AMIN? <SearchName>

Specifies an address, or sets the start value of an address range. The string parameter does not accept the bit value X (don't care).

Parameters:
<Address>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:AMAX <SearchName>,<AddressTo>
SEARch:TRIGger:USB:AMAX? <SearchName>

Sets the the end value of an address range if `TRIGger<m>:USB:ACONdition` is set to `INRange` or `OORange`. The string parameter does not accept the bit value X (don't care).

Parameters:
<AddressTo>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:BITSterror <SearchName>,<BitstuffError>
SEARch:TRIGger:USB:BITSterror? <SearchName>

Defines, whether a search for any bitstuffing error shall be activated or not.

Parameters:
<BitstuffError> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:CRC16error <SearchName>,<CRC16Error>
SEARch:TRIGger:USB:CRC16error? <SearchName>

Defines, whether a search for any CRC16 error shall be activated or not.

Parameters:
<CRC16Error> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:CRC5error <SearchName>,<CRC5Error>
SEARch:TRIGger:USB:CRC5error? <SearchName>

Defines, whether a search for any CRC5 error shall be activated or not.

Parameters:
<CRC5Error> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:DATA <SearchName>,<USBDataType>
SEARch:TRIGger:USB:DATA? <SearchName>

Defines, which data packet type is searched for: "DATA0", "DATA1", "DATA2", or "MDATA", as well as "Any" data packet.

Parameters:
<USBDataType> ANY | D0 | D1 | D2 | MD

ANY
Searches for any of the data packet types listed below

D0
Searches for a DATA0 packet (even PID)

D1
Searches for a DATA1 packet (odd PID)

D2
Searches for a DATA2 packet (high-speed data packet for high bandwidth isochronous transaction in a microframe)

MD
Searches for an MDATA packet (high-speed data packet for split and high bandwidth isochronous transactions)

*RST: D0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:DCONdition <SearchName>,<OperatorData>
SEARch:TRIGger:USB:DCONdition? <SearchName>

Sets the operator (equal or unequal) to set a specific payload data pattern.

Parameters:
<OperatorData> OFF | ANY | EQUal | NEQual

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:DPOPerator <SearchName>,<DataPosOperator>
SEARch:TRIGger:USB:DPOPerator? <SearchName>

Sets the operator (any or equal) for the payload data index position.

Parameters:

<DataPosOperator> ANY | OFF | EQUal

ANY = OFF
The position of payload data is not relevant for the search condition.

*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:DPOSition <SearchName>,<DataPosition>
SEARch:TRIGger:USB:DPOSition? <SearchName>

Specifies the position within a payload data packet, in which a special data pattern is to be searched.

Parameters:

<DataPosition> Range: 0 to 1024
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:ECONdition <SearchName>,<EndpointOptor>
SEARch:TRIGger:USB:ECONdition? <SearchName>

Sets the operator to set a specific endpoint or an endpoint range. The endpoint values are set with `SEARch:TRIGger:USB:EMIN` and `SEARch:TRIGger:USB:EMAX`.

Parameters:

<EndpointOptor> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one endpoint value to be set using `SEARch:TRIGger:USB:EMIN`.

INRange | OORange
In range, out of range. These conditions require a range of endpoint values to be set using `SEARch:TRIGger:USB:EMIN` and `SEARch:TRIGger:USB:EMAX`.

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:EMIN <SearchName>,<Endp>
SEARch:TRIGger:USB:EMIN? <SearchName>

Specifies an endpoint, or sets the start value of an endpoint range.

Parameters:
<Endp>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:EMAX <SearchName>,<EndpTo>
SEARch:TRIGger:USB:EMAX? <SearchName>

Sets the the end value of an endpoint range if `TRIGger<m>:USB:ECONdition` is set to `INRange` or `OORange`.

Parameters:
<EndpTo>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:FCONdition <SearchName>,<FrameNoOperator>
SEARch:TRIGger:USB:FCONdition? <SearchName>

Sets the operator to set a specific frame number or a frame number range. The frame number values are set with `SEARch:TRIGger:USB:FMIN` and `SEARch:TRIGger:USB:FMAX`.

Parameters:
<FrameNoOperator> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one frame number value to be set using `SEARch:TRIGger:USB:FMIN`.

INRange | OORange
In range, out of range. These conditions require a range of frame number values to be set using `SEARch:TRIGger:USB:FMIN` and `SEARch:TRIGger:USB:FMAX`.

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:FMIN <SearchName>,<FrameNumber>
SEARch:TRIGger:USB:FMIN? <SearchName>

Specifies a frame number, or sets the start value of a frame number range.

Parameters:
<FrameNumber>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:FMAX <SearchName>,<FrameNumberTo>
SEARch:TRIGger:USB:FMAX? <SearchName>

Sets the the end value of a frame number range if `TRIGger<m>:USB:FCONdition` is set to `INRange` or `OORange`.

Parameters:
<FrameNumberTo>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:GLITcherror <SearchName>,<GlitchError>
SEARch:TRIGger:USB:GLITcherror? <SearchName>

Defines, whether a search for any glitch error shall be activated or not.

Parameters:
<GlitchError> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:HAND <SearchName>,<HandshakeType>
SEARch:TRIGger:USB:HAND? <SearchName>

Defines, which handshake type is searched for.

Parameters:
<HandshakeType> ANY | ACK | NAK | STALI | NYET
*RST: ACK

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:PATT <SearchName>,<PayloadMuster>
SEARch:TRIGger:USB:PATT? <SearchName>

Defines the payload data pattern to search for.

Parameters:
<PayloadMuster>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:PCONdition <SearchName>,<OperatorPort>
SEARch:TRIGger:USB:PCONdition? <SearchName>

Sets the operator to set a specific port number or a port number range. The port number values are set with `SEARch:TRIGger:USB:PMIN` and `SEARch:TRIGger:USB:PMAX`.

Parameters:
<OperatorPort> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one port number to be set using `SEARch:TRIGger:USB:PMIN`.

INRange | OORange
In range, out of range. These conditions require a port number range to be set using `SEARch:TRIGger:USB:PMIN` and `SEARch:TRIGger:USB:PMAX`.

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:PMIN <SearchName>,<Port>
SEARch:TRIGger:USB:PMIN? <SearchName>

Specifies a port number, or sets the start value of a port number range.

Parameters:
<Port>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:PMAX <SearchName>,<PortTo>
SEARch:TRIGger:USB:PMAX? <SearchName>

Sets the the end value of a port number range if `TRIGger<m>:USB:PCONdition` is set to `INRange` or `OORange`.

Parameters:
<PortTo>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:PIDerror <SearchName>,<PIDError>
SEARch:TRIGger:USB:PIDerror? <SearchName>

Defines, whether a search for any PID error shall be activated or not.

Parameters:
<PIDError> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SCONdition <SearchName>,<OperatorSEU>
SEARch:TRIGger:USB:SCONdition? <SearchName>

Sets the operator to set a specific SEU or an SEU range. The SEU values are set with `SEARch:TRIGger:USB:SMIN` and `SEARch:TRIGger:USB:SMAX`.

For SEU, see "SEU" on page 831.

Parameters:
<OperatorSEU> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one SEU value to be set using `SEARch:TRIGger:USB:SMIN`.

INRange | OORange
In range, out of range. These conditions require a range of SEU values to be set using `SEARch:TRIGger:USB:SMIN` and `SEARch:TRIGger:USB:SMAX`.

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SMIN <SearchName>,<SEU>
SEARch:TRIGger:USB:SMIN? <SearchName>

Specifies an SEU, or sets the start value of an SEU range.

Parameters:
<SEU>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SMAX <SearchName>,<SEUTo>
SEARch:TRIGger:USB:SMAX? <SearchName>

Sets the the end value of an SEU range if `TRIGger<m>:USB:SCONdition` is set to `INRange` or `OORange`.

Parameters:
<SEUTo>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SDATa <SearchName>,<DataPacket>
SEARch:TRIGger:USB:SDATa? <SearchName>

Enables the search for any data packet.

Parameters:
<DataPacket> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SERRor <SearchName>,<ErrorCondition>
SEARch:TRIGger:USB:SERRor? <SearchName>

Enables the search for various errors.

Parameters:
<ErrorCondition> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SHANdshake <SearchName>,<HandshakePacket>
SEARch:TRIGger:USB:SHANdshake? <SearchName>

Defines, whether the search for any handshake packet shall be activated or not.

Parameters:
<HandshakePacket> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SSOP <SearchName>,<StartofPacket>
SEARch:TRIGger:USB:SSOP? <SearchName>

Defines, whether a start of packet (SOP) search shall be activated or not.

Parameters:
<StartofPacket> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SSPE <SearchName>,<SpecialPacket>
SEARch:TRIGger:USB:SSPE? <SearchName>

Defines, whether the search for any special PID packet shall be activated or not.

Parameters:
<SpecialPacket> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:SPEC <SearchName>,<USBSpecialType>
SEARch:TRIGger:USB:SPEC? <SearchName>

Defines, which special PID packet type is searched for: "PREamble", "ERR", "SPLit", "PING", or "Any" special PID packet.

Parameters:
<USBSpecialType> ANY | PREamble | ERR | SPLit | PING

ANY
Searches for any of the special PID packet types listed below

PREamble
Searches for any host-issued preamble token (enables downstream bus traffic to low speed USB devices)

ERR
Searches for any SPLIT transaction error handshake token (reuses PRE value)

SPLit
Searches for any high speed SPLIT transaction token

PING
Searches for any high speed flow control probe for a bulk/control endpoint

*RST: PREamble

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:STCO <SearchName>,<SC>
SEARch:TRIGger:USB:STCO? <SearchName>

Defines, which Start- / Complete- (SC) split transaction type is searched for: SSPLIT or CSPLIT.

Parameters:
<SC> ONE | ZERO | DC

ONE
SC = 1 represents a complete-split (CSPLIT) transaction

ZERO
SC = 0 represents a start-split (SSPLIT) transaction

DC
SC = X represents "don't care" (DC)

*RST: DC

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:STOKen <SearchName>,<TokenPacket>
SEARch:TRIGger:USB:STOKen? <SearchName>

Defines, whether a search for any token packet shall be activated or not.

Parameters:
<TokenPacket> ON | OFF

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:TCONdition <SearchName>,<OperatorET>
SEARch:TRIGger:USB:TCONdition? <SearchName>

Sets the operator to set a specific endpoint type (ET) or an ET range. The ET values are set with `SEARch:TRIGger:USB:TMIN` and `SEARch:TRIGger:USB:TMAX`.

Parameters:
<OperatorET> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

OFF = ANY
The endpoint type is not relevant for the search condition.

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one ET value to be set using `SEARch:TRIGger:USB:TMIN`.

INRange | OORange
In range, out of range. These conditions require a range of ET values to be set using `SEARch:TRIGger:USB:TMIN` and `SEARch:TRIGger:USB:TMAX`.

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:TMIN <SearchName>,<ET>
SEARch:TRIGger:USB:TMIN? <SearchName>

Specifies an endpoint type (ET), or sets the start value of an ET range.

Parameters:
<ET>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:TMAX <SearchName>,<ETTo>
SEARch:TRIGger:USB:TMAX? <SearchName>

Sets the the end value of an endpoint type (ET) range if `TRIGger<m>:USB:TCONdition` is set to `INRange` or `OORange`.

Parameters:
<ETTo>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:TOKen <SearchName>,<USBTokenType>
SEARch:TRIGger:USB:TOKen? <SearchName>

Defines, which token packet type is searched for: "OUT", "IN", "SOF", "SETUP", or "Any" token packet.

Parameters:
<USBTokenType> ANY | OUT | IN | SOF | SETup

ANY
Searches for any of the token packet types listed below.

OUT
Searches for an OUT token.

IN
Searches for an IN token.

SOF
Searches for a start of frame (SOF) token.

SETup
Searches for a SETup token.

*RST: OUT

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WADD <SearchName>,<AddressCheck>
SEARch:TRIGger:USB:WADD? <SearchName>

Defines, whether a search for any address field shall be activated or not.

Parameters:
<AddressCheck> ON | OFF

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WEND <SearchName>,<EndpointCheck>
SEARch:TRIGger:USB:WEND? <SearchName>

Defines, whether a search for any endpoint shall be activated or not.

Parameters:
<EndpointCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WETCheck <SearchName>,<WithETCheck>
SEARch:TRIGger:USB:WETCheck? <SearchName>

Defines, whether a search for any Endpoint Type (ET) shall be activated or not.

Parameters:
<WithETCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WFRN <SearchName>,<FrameNoCheck>
SEARch:TRIGger:USB:WFRN? <SearchName>

Defines, whether a search for any frame number shall be activated or not.

Parameters:
<FrameNoCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WPAY <SearchName>,<PayloadCheck>
SEARch:TRIGger:USB:WPAY? <SearchName>

Defines, whether a search for any payload data shall be activated or not.

Parameters:
<PayloadCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WPID <SearchName>,<WithPIDCheck>
SEARch:TRIGger:USB:WPID? <SearchName>

Defines, whether a search for any packet ID error shall be activated or not.

Parameters:
<WithPIDCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WPOR <SearchName>,<WithPortCheck>
SEARch:TRIGger:USB:WPOR? <SearchName>

Defines, whether a search for any port shall be activated or not.

Parameters:
<WithPortCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WSEU <SearchName>,<WithSEUCheck>
SEARch:TRIGger:USB:WSEU? <SearchName>

Defines, whether a search for any SEU shall be activated or not.

For SEU, see "SEU" on page 831.

Parameters:
<WithSEUCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:USB:WSTC <SearchName>,<WithSCCheck>
SEARch:TRIGger:USB:WSTC? <SearchName>

Defines, whether a search for any Start / Complete (SC) shall be activated or not.

Parameters:
<WithSCCheck> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

24.17.21.5 Search results

In all `SEARch:RESult:USB:PACKet<m>` commands, the suffix <m> selects the packet number in the list of search results.

SEARch:RESult:USB:PACKet<m>:ADDRess?....2183
SEARch:RESult:USB:PACKet<m>:CRC?....2183
SEARch:RESult:USB:PACKet<m>:DATA?....2184
SEARch:RESult:USB:PACKet<m>:ENDPoint?....2184
SEARch:RESult:USB:PACKet<m>:ET?....2184
SEARch:RESult:USB:PACKet<m>:FRAMe?....2184
SEARch:RESult:USB:PACKet<m>:PID?....2185
SEARch:RESult:USB:PACKet<m>:PORT?....2185
SEARch:RESult:USB:PACKet<m>:SC?....2186
SEARch:RESult:USB:PACKet<m>:SEU?....2186
SEARch:RESult:USB:PACKet<m>:STARt?....2186
SEARch:RESult:USB:PACKet<m>:STATus?....2187
SEARch:RESult:USB:PACKet<m>:STOP?....2187
SEARch:RESult:USB:PCOunt?....2187

SEARch:RESult:USB:PACKet<m>:ADDRess? <SearchName>

Returns the packet address for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Addr> Range: 0 to 127
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:USB:PACKet<m>:CRC? <SearchName>

Returns the packet CRC (Cyclic Redundancy Code) for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<CRC> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:USB:PACKet<m>:DATA? <SearchName>

Returns the payload data from the packet with the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:USB:PACKet<m>:ENDPoint? <SearchName>

Returns the endpoint for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Endp>

Range:	0 to 15
Increment:	1
*RST:	0

Usage: Query only

SEARch:RESult:USB:PACKet<m>:ET? <SearchName>

Returns the endpoint type (ET) for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<ET>

Range:	0 to 3
Increment:	1
*RST:	0

Usage: Query only

SEARch:RESult:USB:PACKet<m>:FRAMe? <SearchName>

Returns the frame number for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameNo> Range: 0 to 2047
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:USB:PACKet<m>:PID? <SearchName>

Returns the packet PID for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<PID> RES | OUT | ACK | DATA0 | DATa0 | PING | SOF | NYET | DATA2 | DATa2 | SPLIT | SPLit | IN | NAK | DATA1 | DATa1 | PRE | SETUP | SETup | STALL | STALI | MDATA | MDATa | UNK

For a description of the return values, see `BUS<m>:USB:PACKet<n>:PID?` on page 2163.

*RST: RES

Usage: Query only

SEARch:RESult:USB:PACKet<m>:PORT? <SearchName>

Returns the port number for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Port> Range: 0 to 127
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:USB:PACKet<m>:SC? <SearchName>

Returns the Start- / Complete-split transaction (SSPLIT / CSPLIT) flag bits for the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SC>

Usage: Query only

SEARch:RESult:USB:PACKet<m>:SEU? <SearchName>

Returns the SEU values for the selected packet number within the search result.

For SEU, see "SEU" on page 831.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<SEU>

Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:USB:PACKet<m>:STARt? <SearchName>

Returns the start time of the packet with the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Start>

Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:USB:PACKet<m>:STATus? <SearchName>

Returns the status of the packet with the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> OK | PID | CRC | BTST | GLITCH | GLITch | BYTE
*RST: OK

Usage: Query only

SEARch:RESult:USB:PACKet<m>:STOP? <SearchName>

Returns the stop time of the packet with the selected packet number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:USB:PCOunt? <SearchName>

Returns the search result's packet count, i.e. the number of packets found in the search result.

Query parameters:
<SearchName>

Return values:
<Count> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

24.17.22 USB 3.1 (option R&S RTO6-K570)

- Configuration....2188
- Digital signal processing (DSP)....2190
- Trigger....2193
- Decode results....2204
- Search settings....2208
- Search results....2214

24.17.22.1 Configuration

In all `BUS<m>:USBT` commands, the suffix <m> selects the serial bus.

BUS<m>:USBThree:SOURce....2188
BUS<m>:USBThree:POLarity....2188
BUS<m>:USBThree:THRHigh....2189
BUS<m>:USBThree:THRLow....2189
BUS<m>:USBThree:SCRMode....2189
BUS<m>:USBThree:TYPE....2189

BUS<m>:USBThree:SOURce <SourceData>

Selects the source for the data signal in the USB 3.1 protocol.

Suffix:
<m> 1..4

Parameters:
<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: C1W1

Usage: Asynchronous command

BUS<m>:USBThree:POLarity <Polarity>

Defines the polarity of the data signal for the active (U0) state of the port.

Suffix:
<m> 1..4

Parameters:
<Polarity> NORMal | INVert

NORMal
Normal (non-inverted) data signal polarity

INVert
Inverted data signal polarity

*RST: NORMal

BUS<m>:USBThree:THRHigh <ThresholdUpper>

Defines the upper threshold level for digitization of the data signal in the USB 3.1 protocol.

Suffix:
<m> 1..4

Parameters:
<ThresholdUpper> Range: -2 to 2
Increment: 0.01
*RST: 0.05
Default unit: V

BUS<m>:USBThree:THRLow <ThresholdLower>

Defines the lower threshold level for digitization of the data signal in the USB 3.1 protocol.

Suffix:
<m> 1..4

Parameters:
<ThresholdLower> Range: -2 to 2
Increment: 0.01
*RST: -0.05
Default unit: V

BUS<m>:USBThree:SCRMode <ScrambleMode>

Selects if the data pattern is scrambled or unscambled.

Suffix:
<m> 1..4

Parameters:
<ScrambleMode> SCRM | NSCR

SCRM
Scrambling is enabled as specified in the standard, to minimize EMI emissions.

NSCR
No scrambling: To simplify testing and debugging, scrambling is disabled.

*RST: SCRM

BUS<m>:USBThree:TYPE <Protocol>

Selects the USB 3.1 protocol type.

Suffix:
<m> 1..4

Parameters:
<Protocol> SS
*RST: SS

Usage: Asynchronous command

24.17.22.2 Digital signal processing (DSP)

BUS<m>:CTLE:DCGain....2190
BUS<m>:CTLE:ENBLe....2190
BUS<m>:CTLE:P1FReq....2190
BUS<m>:CTLE:P2FReq....2191
BUS<m>:CTLE:Z1FReq....2191
BUS<m>:USBThree:BITR....2191
BUS<m>:USBThree:DFE:COEFficient....2191
BUS<m>:USBThree:DFE:ENBLe....2192
BUS<m>:USBThree:DFE:ESTCoeff....2192
BUS<m>:USBThree:DFE:TCONt....2192
BUS<m>:USBThree:PROTocol....2192

BUS<m>:CTLE:DCGain <DCGain>

Sets the adjustable DC gain. This value can be chosen according to the channel loss characteristics. For low-loss channels, you can set a value close to 0 dB.

Suffix:
<m> 1..4

Parameters:
<DCGain> Range: -100 to 100
Increment: 0.01
*RST: 0
Default unit: dB

BUS<m>:CTLE:ENBLe <EnableCTLE>

Enables the continuous time linear equalization (CTLE) equalization.

Suffix:
<m> 1..4

Parameters:
<EnableCTLE> ON | OFF
*RST: OFF

BUS<m>:CTLE:P1FReq <PoleOneFreq>

Sets the pole 1 frequency for the transfer function of the CTLE algorithm.

Suffix:
<m> 1..4

Parameters:

<PoleOneFreq>	Range:	1 to 40E+9
	Increment:	1
	*RST:	0
	Default unit:	Hz

BUS<m>:CTLE:P2FReq <PoleTwoFreq>

Sets the pole 2 frequency for the transfer function of the CTLE algorithm.

Suffix:

<m>	1..4

Parameters:

<PoleTwoFreq>	Range:	1 to 40E+9
	Increment:	1
	*RST:	0
	Default unit:	Hz

BUS<m>:CTLE:Z1FReq <ZroOneFreq>

Sets the zero frequency for the transfer function of the CTLE algorithm.

Suffix:

<m>	1..4

Parameters:

<ZroOneFreq>	Range:	1 to 40E+9
	Increment:	1
	*RST:	0
	Default unit:	Hz

BUS<m>:USBThree:BITR <HighSpdBitrate>

Sets the bit rate.

Suffix:

<m>	1..4

Parameters:

<HighSpdBitrate>	Range:	1000 to 3E+9
	Increment:	1000
	*RST:	3E+9
	Default unit:	bps

BUS<m>:USBThree:DFE:COEFficient <DFEcoeff>

Sets the value of the coefficient for the DFE .

Suffix:

<m>	1..4

Parameters:

<DFEcoeff>	Range:	-1 to 1
	Increment:	1E-3
	*RST:	0
	Default unit:	V

BUS<m>:USBThree:DFE:ENBLe <EnableDFE>

Enables the use of the decision feedback equalization for USB 3.1.

Suffix:

<m>	1..4

Parameters:

<EnableDFE>	ON | OFF	
	*RST:	ON

BUS<m>:USBThree:DFE:ESTCoeff <AutoChEstimate>

Enables the use of estimate coefficients for the decision feedback equalization.

Suffix:

<m>	1..4

Parameters:

<AutoChEstimate>	ON | OFF	
	*RST:	ON

BUS<m>:USBThree:DFE:TCONt <CIRLength>

Sets a number of taps for the decision feedback filter.

Suffix:

<m>	1..4

Parameters:

<CIRLength>	Range:	1 to 20
	Increment:	1
	*RST:	5

BUS<m>:USBThree:PROTocol <ProtHSSPTSt>

Selects the USB Super Speed channel length.

Suffix:

<m>	1..4

Parameters:

<ProtHSSPTSt>	OFF | LOWFLTR | HIGHFLTR	
	*RST:	LOWFLTR

24.17.22.3 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

There are two commands for each parameter, that you can use for defining the USB 3.1 settings.

For example, to set the *Frame type =ACK > Field =Route String >Data* value you can use one of the following commands:

- `TRIGger:USBThree:FRAMe13:FLD1:DMIN 01100`
  Defines the parameter by using the index <m> for the frame number and <n> for the field number.
- `TRIGger:USBThree:DMIN "ACK", "Route String", 01100`
  Defines the parameter by using the frame and field name.

Table 24-14: Overview of frame and field numbers for USB 3.1, Protocol Type > Super Speed

Frame number	Frame name	Field number	Field name
1	TSEQ	1	Symbol
2	TS1	1	Train
		2	Loop
		3	Scramble
		4	LL
		5	Retimer
3	TS1A	-	-
4	TS1B	-	-
5	TS2	1	Train
		2	Loop
		3	Scramble
		4	LL
		5	Retimer
6	Set Link Function	1	SLF
		2	HDr #
		3	Hub #
		4	DLY
		5	DFR

Frame number	Frame name	Field number	Field name
7	U2 Inactivity Timeout	1	U2 IT
		2	HDr #
		3	Hub #
		4	DLY
		5	DFR
8	Vendor Device Test	1	VDT
		2	Vendor Def
		3	HDr #
		4	Hub #
		5	DLY
		6	DFR
9	Port Capability	1	Link Speed
		2	Num HP Buf
		3	Dir
		4	OTG
		5	Tiebreake
		6	HDr #
		7	Hub #
		8	DLY
		9	DFR
10	Port Config	1	Link Speed
		2	HDr #
		3	Hub #
		4	DLY
		5	DFR
11	Port Config Rsp	1	Response Code
		2	HDr #
		3	Hub #
		4	DLY
		5	DFR
12	Link Delay Meas	1	LDM Type
		2	LDMS
		3	Response Delay

Frame number	Frame name	Field number	Field name
		4	HDr #
		5	Hub #
		6	DLY
		7	DFR
13	ACK	1	Route String
		2	Device Address
		3	rty
		4	D
		5	EPTNum
		6	TT
		7	HE
		8	NumP
		9	SeqNum
		10	TPF
		11	StreamId
		12	SSI
		13	WPA
		14	DBI
		15	PP
		16	NBI
		17	HDr #
		18	Hub #
		19	DLY
		20	DFR
14	NRDY	1	Device Address
		2	D
		3	EPTNum
		4	StreamId
		5	HDr #
		6	Hub #
		7	DLY
		8	DFR
15	ERDY	1	Device Address

Frame number	Frame name	Field number	Field name
		2	D
		3	EPTNum
		4	NumP
		5	StreamId
		6	HDr #
		7	Hub #
		8	DLY
		9	DFR
16	Status	1	Route String
		2	Device Address
		3	D
		4	EPTNum
		5	PP
		6	HDr #
		7	Hub #
		8	DLY
		9	DFR
17	Stall	1	Device Address
		2	D
		3	EPTNum
		4	HDr #
		5	Hub #
		6	DLY
		7	DFR
18	Function Wake	1	Device Address
		2	Interface
		3	HDr #
		4	Hub #
		5	DLY
		6	DFR
19	Latency Tolerance	1	Device Address
		2	BELT
		3	HDr #

Frame number	Frame name	Field number	Field name
		4	Hub #
		5	DLY
		6	DFR
20	Bus Interval Adjust	1	Device Address
		2	BusIntAdj
		3	HDr #
		4	Hub #
		5	DLY
		6	DFR
21	Host Role Req	1	Device Address
		2	RSP
		3	HDr #
		4	Hub #
		5	DLY
		6	DFR
22	Sublink Speed	1	Device Address
		2	TPF
		3	LSE
		4	ST
		5	Lanes
		6	LP
		7	LSM
		8	HDr #
		9	Hub #
		10	DLY
		11	DFR
23	Ping	1	Route String
		2	Device Address
		3	D
		4	EPTNum
		5	HDr #
		6	Hub #
		7	DLY

Frame number	Frame name	Field number	Field name
		8	DFR
24	Ping Rsp	1	Device Address
		2	D
		3	EPTNum
		4	HDr #
		5	Hub #
		6	DLY
		7	DFR
25	Data Packet Header	1	Route String
		2	Device Address
		3	SeqNum
		4	EOB
		5	D
		6	EPTNum
		7	TT
		8	S
		9	Length
		10	StreamId
		11	SSI
		12	WPA
		13	DBI
		14	PP
		15	NBI
		16	HDr #
		17	Hub #
		18	DLY
		19	DFR
26	Data Packet Payload	1	Data
27	DPP Aborted	1	Data
28	ITP	1	Counter
		2	Delta
		3	Adj Ctrl
		4	Corr

Frame number	Frame name	Field number	Field name
		5	HDr #
		6	Hub #
		7	DLY
		8	DFR
29	Link Command	1	Sub-type
30	Skip	-	-
31	BRST	-	-
32	BDAT	-	-
33	BERC	-	-
34	BCNT	1	EC
35	Idle	-	-

TRIGger<m>:USBThree:ERENable 2199
TRIGger<m>:USBThree:ERRor<n>:ENABle 2199
TRIGger<m>:USBThree:FRENable 2200
TRIGger<m>:USBThree:FRAMe<n>:ENABle 2200
TRIGger<m>:USBThree:FIENable 2200
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:ENABle 2200
TRIGger<m>:USBThree:DOPerator 2201
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DOPerator 2201
TRIGger<m>:USBThree:DMIN 2201
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DMIN 2201
TRIGger<m>:USBThree:DMAX 2202
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DMAX 2202
TRIGger<m>:USBThree:BIT 2202
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:BIT 2202
TRIGger<m>:USBThree:IOPerator 2202
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IOPerator 2202
TRIGger<m>:USBThree:IMIN 2203
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IMIN 2203
TRIGger<m>:USBThree:IMAX 2203
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IMAX 2203

TRIGger<m>:USBThree:ERENable <ErrorName>, <Enabler>
TRIGger<m>:USBThree:ERRor<n>:ENABle <Enable>

Defines the error type to be triggered on. You can trigger on all error types in parallel.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the error number.

Parameters:

<Enable> ON | OFF

*RST: ON

Example: Trigger on CRC and length errors:

`TRIGger:USBThree:ERRor1:ENABle ON`

Enables CRC Error.

`TRIGger:USBThree:ERENable "Length Error",ON`

Enables Length Error.

TRIGger<m>:USBThree:FRENable <Frame>, <Enabler>
TRIGger<m>:USBThree:FRAMe<n>:ENABle <Enable>

Enables or disables the checking condition for the selected frame.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the frame number.

Parameters:

<Enable> ON | OFF

*RST: OFF

Example: `TRIGger:USBThree:FRAMe1:ENABle ON`

Enables the checking condition for the TSEQ frame.

`TRIGger:USBThree:FRENable "TS1" ON`

Enables the checking condition for the TS1 frame.

TRIGger<m>:USBThree:FIENable <Frame>, <Field>, <Enabler>
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:ENABle <CondEnabler>

Enables or disables the checking condition for the selected field of the selected frame.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the frame number.

<o> *
Specifies the field number within the frame.

Parameters:

<CondEnabler> ON | OFF

ON
Checking condition enabled

OFF
Checking condition disabled

*RST: OFF

Example: `TRIGger:USBThree:FRAMe1:FLD1:ENABle ON`
Enables the checking condition for the Symbol field of the TSEQ frame.
`TRIGger:USBThree:FRENable "TS1", "Loop" ON`
Enables the checking condition for the Loop field of the TS1 frame.

TRIGger<m>:USBThree:DOPerator <Frame>, <Field>, <Operator>
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DOPerator <DataOperator>

Sets the operator for the data pattern in the selected field of the selected frame.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the field number within the frame.

<o> *

Parameters:
<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one endpoint value to be set using `TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DMIN`.

INRange | OORange
In range, out of range. These conditions require a range of endpoint values to be set using `TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DMIN` and `TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DMAX`.

*RST: EQUal

TRIGger<m>:USBThree:DMIN <Frame>, <Field>, <Data>
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DMIN <DataMin>

Specifies the data pattern, or sets the start value of a data pattern range.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the field number within the frame.

<o> *

Parameters:
<DataMin>

TRIGger<m>:USBThree:DMAX <Frame>, <Field>, <Data>
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DMAX <DataMax>

Sets the end value of a data pattern range if `TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DOPerator` is set to `INRange` or `OORange`.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the field number within the frame.

<o> *

Parameters:

<DataMax>

TRIGger<m>:USBThree:BIT <Frame>, <Field>, <Bit>
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:BIT <BitState>

Sets the bit state of a field that only consists of one bit.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the field number within the frame.

<o> *

Parameters:

<BitState> ONE | ZERO | DC

ONE
1

ZERO
0

DC
"Don't care" (DC) = X

*RST: DC

TRIGger<m>:USBThree:IOPerator <Frame>, <Field>, <Operator>
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IOPerator <IndexOperator>

Sets the operator for the index in the selected field of the selected frame.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the field number within the frame.

<o> *

Parameters:

<IndexOperator> EQUal | INRange | RANGe

EQUal
This condition requires one endpoint value to be set using `TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IMIN` and `TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IMAX`.

*RST: INRange

TRIGger<m>:USBThree:IMIN <Frame>, <Field>, <Data>
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IMIN <IndexMin>

Specifies the index, or sets the start value of an index range.

Suffix:

<m> 1..3

<n> *
Specifies the field number within the frame.

<o> *

Parameters:

<IndexMin> Specifies the index value or index start value within the field. The index range, increment and *RST values depend on the field type.

TRIGger<m>:USBThree:IMAX <Frame>, <Field>, <Data>
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IMAX <IndexMax>

Sets the end value of an index range if `TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IOPerator` is set to `INRange` or `RANGe`.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the field number within the frame.

<o> *

Parameters:

<IndexMax> Specifies the end value for the index range within the field. The index range, increment and *RST values depend on the field type.

24.17.22.4 Decode results

In all `BUS<m>:USBThree:RESult:FRAMe<n>` commands, the suffix <m> selects the serial bus and the suffix <n> selects the frame number in the decode table.

BUS<m>:USBThree:RESult:FCOunt?.....2204
BUS<m>:USBThree:RESult:FRAMe<n>:INFO?.....2204
BUS<m>:USBThree:RESult:FRAMe<n>:TYPE?.....2204
BUS<m>:USBThree:RESult:FRAMe<n>:STATe?.....2205
BUS<m>:USBThree:RESult:FRAMe<n>:STARt?.....2205
BUS<m>:USBThree:RESult:FRAMe<n>:STOP?.....2206
BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:TYPE?.....2206
BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:STATus?.....2206
BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:STARt?.....2207
BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:STOP?.....2207
BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:VAL?.....2207
BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:FVAL?.....2208

BUS<m>:USBThree:RESult:FCOunt?

Returns the number of decoded frames in the selected serial bus.

Suffix:
<m> 1..4

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1

Usage: Query only

BUS<m>:USBThree:RESult:FRAMe<n>:INFO?

Returns specific frame information details of the selected frame. This information is also shown in the "Info" column of the decode results table.

Suffix:
<m> 1..4

<n> *
Specifies the frame number.

Return values:
<Info>

Usage: Query only

BUS<m>:USBThree:RESult:FRAMe<n>:TYPE?

Returns the frame type for the selected serial bus and frame number.

Suffix:
<m> 1..4

<n> *
Selects the frame number.

Return values:

<FrameType> TSEQ | TS1 | TS1A | TS1B | TS2 | SLF | U2IT | VDT | PCAP | PCNF | PCNR | LDM | ACK | NRDY | ERDY | STATus | STALI | FWAKe | LTM | BIAM | HRR | SSPD | PING | PNGR | DPH | DPHN | DPP | DPPA | ITP | LC | SKIP | BRST | BDAT | BERC | BCNT | IDLE | CP0 | CP1 | CP2 | CP3 | CP5or6 | UHP | ULMP | UTP | UDVN | UDEF | ERRor | CTLB | DATB | SYNC | SDS

For a description of the frame types, see "Frame types" on page 842.

*RST: UDEF

Usage: Query only

BUS<m>:USBThree:RESult:FRAMe<n>:STATe?

Returns the status of the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

Return values:

<State> OK | INComplete | UNCorrelated | UNKNown | AMB | CRC | LENGth | VOOR | SYMB

VOOR
Value out of range

Usage: Query only

BUS<m>:USBThree:RESult:FRAMe<n>:STARt?

Returns the start time of the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
Default unit: s

Usage: Query only

BUS<m>:USBThree:RESult:FRAMe<n>:STOP?

Returns the stop time of the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
Default unit: s

Usage: Query only

BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:TYPE?

Returns the type of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<FieldType>

Usage: Query only

BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:STATus?

Returns the status of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<FieldState> OK | INComplete | UNCorrelated | UNKNown | AMB | CRC | LENGth | VOOR | SYMB

VOOR
Value out of range

Usage: Query only

BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:STARt?

Returns the start time of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

<o> *

Return values:

<Start>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:STOP?

Returns the stop time of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

<o> *

Return values:

<Stop>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:VAL?

Returns the value of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:
<FieldValue>

Usage: Query only

BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:FVAL?

Returns the formatted value of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:
<ValueFormatted>

Usage: Query only

24.17.22.5 Search settings

There are two commands for each parameter, that you can use for defining the SSIC search settings.

For example, to set the *Frame type =ACK > Field =Route String >Data* value you can use one of the following commands:

- `SEARch:TRIGger:USBThree:FRAMe13:FLD1:DMIN 01100`
  Defines the parameter by using the index <m> for the frame number and <n> for the field number. For an overview, see Table 24-14.
- `SEARch:TRIGger:USBThree:DMIN "ACK", "Route String", 01100`
  Defines the parameter by using the frame and field name.

SEARch:TRIGger:USBThree:ERENable....2209
SEARch:TRIGger:USBThree:ERRor<m>:ENABle....2209
SEARch:TRIGger:USBThree:FRENable....2209
SEARch:TRIGger:USBThree:FRAMe<m>:ENABle....2209
SEARch:TRIGger:USBThree:FIENable....2210
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:ENABle....2210
SEARch:TRIGger:USBThree:DOPerator....2210
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DOPerator....2210

SEARch:TRIGger:USBThree:DMIN....2211
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DMIN....2211
SEARch:TRIGger:USBThree:DMAX....2211
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DMAX....2211
SEARch:TRIGger:USBThree:BIT....2212
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:BIT....2212
SEARch:TRIGger:USBThree:IOPerator....2212
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IOPerator....2212
SEARch:TRIGger:USBThree:IMIN....2213
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMIN....2213
SEARch:TRIGger:USBThree:IMAX....2213
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMAX....2213

SEARch:TRIGger:USBThree:ERENable <SearchName>,<ErrorName>, <Enabler>
SEARch:TRIGger:USBThree:ERRor<m>:ENABle <SearchName>,<Enable>
SEARch:TRIGger:USBThree:ERRor<m>:ENABle? <SearchName>

Enables the search for errors. You can search for all error types in parallel.

Suffix:
<m> *
Specifies the error.

Parameters:
<Enable> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName> String with the name of the search.

Example: Searches for CRC and length errors:
`SEARch:TRIGger:USBThree:ERRor1:ENABle ON`
Enables searching for CRC Error.
`SEARch:TRIGger:USBThree:ERENable "Length Error",ON`
Enables searching for Length Error.

SEARch:TRIGger:USBThree:FRENable <SearchName>,<Frame>, <Enabler>
SEARch:TRIGger:USBThree:FRAMe<m>:ENABle <SearchName>,<Enable>
SEARch:TRIGger:USBThree:FRAMe<m>:ENABle? <SearchName>

Defines the frame type to be searched for.

Suffix:
<m> *

Parameters:
<Enable> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

Example: `SEARch:TRIGger:USBThree:FRAMe1:ENABle ON`
Enables searching for the TSEQ frame.
`SEARch:TRIGger:USBThree:FRENable "TS1" ON`
Enables searching for the TS1 frame.

SEARch:TRIGger:USBThree:FIENable <SearchName>,<Frame>, <Field>, <Enabler>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:ENABle <SearchName>,<CondEnabler>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:ENABle? <SearchName>

Enables or disables the checking condition for searching a specific field of the selected frame.

Suffix:
<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:
<CondEnabler> ON | OFF

ON
Checking condition enabled

OFF
Checking condition disabled

*RST: OFF

Parameters for setting and query:
<SearchName> String with the name of the search.

Example: `SEARch:TRIGger:USBThree:FRAMe1:FLD1:ENABle ON`
Enables searching for the Symbol field of the TSEQ frame.
`SEARch:TRIGger:USBThree:FRENable "TS1", "Loop" ON`
Enables searching for the Loop field of the TS1 frame.

SEARch:TRIGger:USBThree:DOPerator <SearchName>,<Frame>, <Field>, <Operator>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DOPerator <SearchName>,<DataOperator>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DOPerator? <SearchName>

Sets the operator to set a specific data pattern to be searched in the selected field of the selected frame.

Suffix:
<m> *
Specifies the field number within the frame.

<n> *

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one endpoint value to be set using `SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMIN`.

INRange | OORange
In range, out of range. These conditions require a range of endpoint values to be set using `SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMIN` and `SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMAX` on page 2213.

*RST: EQUal

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:USBThree:DMIN <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DMIN <SearchName>,<DataMin>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DMIN? <SearchName>

Specifies the data pattern to be searched, or sets the start value of a data pattern range to be searched.

Suffix:

<m> *
Specifies the field number within the frame.

<n> *

Parameters:

<DataMin> Specifies the name of the generic trigger setting frame, see `SEARch:RESult:USBThree:FRAMe<m>:TYPE?` on page 2214.

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:USBThree:DMAX <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DMAX <SearchName>,<DataMax>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DMAX? <SearchName>

Sets the end value of a data pattern range if `SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DOPerator` is set to `INRange` or `OORange`.

Suffix:

<m> *
Specifies the field number within the frame.

<n> *

Parameters:

<DataMax> Specifies the name of the generic trigger setting frame, see `SEARch:RESult:USBThree:FRAMe<m>:TYPE?` on page 2214.

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:USBThree:BIT <SearchName>,<Frame>, <Field>, <Bit>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:BIT <SearchName>,<BitState>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:BIT? <SearchName>

Sets the bit state of a field to be searched that only consists of one bit.

Suffix:

<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:

<BitState> ONE | ZERO | DC

ONE
1

ZERO
0

DC
"Don't care" (DC) = X

*RST: DC

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:USBThree:IOPerator <SearchName>,<Frame>, <Field>, <Operator>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IOPerator <SearchName>,<IndexOperator>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IOPerator? <SearchName>

Sets the operator for the index for searching in the selected field of the selected frame.

Suffix:

<m> *
Specifies the field number within the frame.

<n> *

Parameters:

<IndexOperator> EQUal | INRange | RANGe

EQUal
This condition requires one endpoint value to be set using `SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMIN` and `SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMAX`.

*RST: INRange

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:USBThree:IMIN <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMIN <SearchName>,<IndexMin>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMIN? <SearchName>

Specifies the index at which the data is to be searched, or sets the start value of an index range in which the data is to be searched.

Suffix:

<m> *
Specifies the field number within the frame.

<n> *

Parameters:

<IndexMin> Specifies the index value or the start value of an index range within the field.
The index range, increment and *RST values depend on the field type.

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:USBThree:IMAX <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMAX
<SearchName>,<IndexMax>
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMAX? <SearchName>

Sets the end value of an index range if `SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IOPerator` is set to `INRange` or `RANGe`.

Suffix:

<m> *
Specifies the field number within the frame.

<n> *

Parameters:

<IndexMax> Specifies the end value for the index range within the field. The index range, increment and *RST values depend on the field type.

Parameters for setting and query:

<SearchName> String with the name of the search.

24.17.22.6 Search results

In all `SEARch:RESult:USBThree:FRAMe<m>` commands, the suffix <m> selects the frame number in the list of search results.

SEARch:RESult:USBThree:FRAMe<m>:TYPE? 2214
SEARch:RESult:USBThree:FRAMe<m>:INFO? 2214
SEARch:RESult:USBThree:FCOunt? 2215
SEARch:RESult:USBThree:FRAMe<m>:STATe? 2215
SEARch:RESult:USBThree:FRAMe<m>:STARt? 2215
SEARch:RESult:USBThree:FRAMe<m>:STOP? 2216
SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:TYPE? 2216
SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:STATus? 2216
SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:STARt? 2217
SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:STOP? 2217
SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:VAL? 2218
SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:FVAL? 2218

SEARch:RESult:USBThree:FRAMe<m>:TYPE? <SearchName>

Returns the frame type for the selected frame number within the search result.

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<FrameType> TSEQ | TS1T | TS2T | SLF | U2IT | VDT | PCAP | PCNF | PCNR | LDM | ACK | NRDY | ERDY | STATus | STALI | FWAKe | LTM | BIAM | HRR | SSPD | PING | PNGR | DPH | DPP | DPPA | ITP | LC | SKIP | BRST | BDAT | BERC | BCNT | IDLE | UHP | ULMP | UTP | UDVN | UDEF | ERRor

For a description of the frame types, see "Frame types" on page 842.

Usage: Query only

SEARch:RESult:USBThree:FRAMe<m>:INFO? <SearchName>

Returns specific frame information details of the selected frame in the search result. This information is also shown in the "Info" column of the decode results table.

Suffix:
<m> *
Specifies the frame number.

Query parameters:
<SearchName>

Return values:
<Info>

Usage: Query only

SEARch:RESult:USBThree:FCOunt? <SearchName>

Returns the number of decoded frames in the search result.

Query parameters:
<SearchName>

Return values:
<FrameCount>

Usage: Query only

SEARch:RESult:USBThree:FRAMe<m>:STATe? <SearchName>

Returns the status of the frame with the selected frame number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> OK | INComplete | UNCorrelated | UNKNown | AMB | CRC | LENGth | VOOR | SYMB

VOOR
Value out of range

Usage: Query only

SEARch:RESult:USBThree:FRAMe<m>:STARt? <SearchName>

Returns the start time of the frame with the selected frame number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
Default unit: s

Usage: Query only

SEARch:RESult:USBThree:FRAMe<m>:STOP? <SearchName>

Returns the stop time of the frame with the selected frame number within the search result.

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
Default unit: s

Usage: Query only

SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:TYPE? <SearchName>

Returns the field name of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:

<m> *

<n> *
Selects the field number.

Query parameters:

<SearchName>

Return values:

<FieldType>

Usage: Query only

SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:STATus? <SearchName>

Returns the field status of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:

<m> *

<n> *
Selects the field number.

Query parameters:
<SearchName>

Return values:
<FieldState> OK | INComplete | UNCorrelated | UNKNown | AMB | CRC | LENGth | VOOR | SYMB

VOOR
Value out of range

Usage: Query only

SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:STARt? <SearchName>

Returns the start time of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:STOP? <SearchName>

Returns the stop time of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:VAL? <SearchName>

Returns the value of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *
Selects the field number.

Query parameters:
<SearchName>

Return values:
<ByteValue>

Usage: Query only

SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:FVAL? <SearchName>

Returns the formatted value of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *
Selects the field number.

Query parameters:
<SearchName>

Return values:
<ValueFormatted>

Usage: Query only

24.17.23 USBPD (option R&S RTO6-K570)

- Configuration....2218
- Trigger....2220
- Decode results....2227
- Search settings....2232
- Search results....2237

24.17.23.1 Configuration

In all `BUS<m>:USBPd` commands, the suffix <m> selects the serial bus.

BUS<m>:USBPd:DETail....2219
BUS<m>:USBPd:HYSTeresis....2219
BUS<m>:USBPd:SOURce....2219

BUS<m>:USBPd:THReshold 2219
BUS<m>:USBPd:THRBottom 2220
BUS<m>:USBPd:THRMid 2220
BUS<m>:USBPd:THRTop 2220

BUS<m>:USBPd:DETail <Detail>

If enabled, the data words are broken down into subframes. If not enabled the data words are displayed as 32-bit data words.

Suffix:
<m> 1..4

Parameters:
<Detail> ON | OFF
*RST: OFF

BUS<m>:USBPd:HYSTeresis <Hysteresis>

Sets a value for the hysteresis of the data.

Suffix:
<m> 1..4

Parameters:
<Hysteresis> Range: -10 to 10
Increment: 0.01
*RST: 0.05
Default unit: V

BUS<m>:USBPd:SOURce <SourceData>

Selects the source for the data signal in the USBPD protocol.

For triggering on a serial bus, analog input channels are required.

Suffix:
<m> 1..4

Parameters:
<SourceData> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4
*RST: C1W1

Usage: Asynchronous command

BUS<m>:USBPd:THReshold <Threshold>

Sets the threshold value of the data.

Suffix:
<m> 1..4

Parameters:

<Threshold>	Range:	-10 to 10
	Increment:	0.01
	*RST:	0.6
	Default unit:	V

BUS<m>:USBPd:THRBottom <ThresholdBot>

Sets the bottom threshold (for the low current) for the current advertisment mode.

Suffix:

<m>	1..4

Parameters:

<ThresholdBot>	Range:	-10 to 10
	Increment:	0.01
	*RST:	0.2
	Default unit:	V

BUS<m>:USBPd:THRMid <ThresholdMid>

Sets the middle threshold (at default USB Type-C currnet) for the current advertisment mode.

Suffix:

<m>	1..4

Parameters:

<ThresholdMid>	Range:	-10 to 10
	Increment:	0.01
	*RST:	0.66
	Default unit:	V

BUS<m>:USBPd:THRTop <ThresholdTop>

Sets the top threshold (at USB Type-C currnet of 1.5 A) for the current advertisment mode.

Suffix:

<m>	1..4

Parameters:

<ThresholdTop>	Range:	-10 to 10
	Increment:	0.01
	*RST:	1.23
	Default unit:	V

24.17.23.2 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

There are two commands for each parameter, that you can use for defining the USBPD settings.

For example, to set the *Frame type =Control Frame > Field =MsgID >Data* value you can use one of the following commands:

- `TRIGger:USBPD:FRAMe3:FLD3:DMIN 01100`
  Defines the parameter by using the index <m> for the frame number and <n> for the field number. For an overview, see Overview of frame and field numbers for USBPD.
- `TRIGger:USBPD:DMIN "Control", "MsgID", 01100`
  Defines the parameter by using the frame and field name.

Table 24-15 gives an overview of the available frame and field numbers. Those numbers follow the GUI numbers.

Table 24-15: Overview of frame and field numbers for USBPD

Frame number	Frame name	Field number	Field name
1	Test Frame	-	-
2	Reset	-	-
3	Control frame	1	Extended
		2	NumDataObjs
		3	MsgID
		4	PwrRole/Plug
		5	Rev
		6	DataRole
		7	MsgType
4	Data Frame	1	Extended
		2	NumDataObjs
		3	MsgID
		4	PwrRole/Plug
		5	Rev
		6	DataRole
		7	MsgType
		8	DATA
5	Extended	1	Extended

Frame number	Frame name	Field number	Field name
		2	NumDataObjs
		3	MsgID
		4	PwrRole/Plug
		5	Rev
		6	DataRole
		7	MsgType
		8	Chunked
		9	Chunk Num
		10	Req Chunk
		11	Data Size
		12	DATA

TRIGger<m>:USBPd:ERENable....2222
TRIGger<m>:USBPd:ERRor<n>:ENABle....2222
TRIGger<m>:USBPd:FRENable....2223
TRIGger<m>:USBPd:FRAMe<n>:ENABle....2223
TRIGger<m>:USBPd:FIENable....2223
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:ENABle....2223
TRIGger<m>:USBPd:BIT....2224
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:BIT....2224
TRIGger<m>:USBPd:DMAX....2224
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMAX....2224
TRIGger<m>:USBPd:DMIN....2225
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMIN....2225
TRIGger<m>:USBPd:DOPerator....2225
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DOPerator....2225
TRIGger<m>:USBPd:IMAX....2225
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMAX....2225
TRIGger<m>:USBPd:IMIN....2226
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMIN....2226
TRIGger<m>:USBPd:IOPerator....2226
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IOPerator....2226

TRIGger<m>:USBPd:ERENable <ErrorName>, <Enabler>
TRIGger<m>:USBPd:ERRor<n>:ENABle <Enable>

Enables or disables the checking condition for a specific error in the selected field of the selected frame.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the error number.

Parameters:
<Enable> ON | OFF
*RST: ON

Example: `TRIGger:USBPd:ERRor1:ENABle ON`
Enables CRC Error.
`TRIGger:USBPd:ERENable "Length Error",ON`
Enables Length Error.

TRIGger<m>:USBPd:FRENable <Frame>, <Enabler>
TRIGger<m>:USBPd:FRAMe<n>:ENABle <Enable>

Enables or disables the checking condition for the selected frame.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the frame number.

Parameters:
<Enable> ON | OFF
*RST: OFF

Example: `TRIGger:USBPd:FRAMe1:ENABle ON`
Enables searching for for the Test frame.
`TRIGger:USBPd:FRENable "Control Frame" ON`
Enables searching for the Control frame.

TRIGger<m>:USBPd:FIENable <Frame>, <Field>, <Enabler>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:ENABle <CondEnabler>

Enables or disables the checking condition for a specific data pattern in the selected field of the selected frame.

Suffix:
<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the frame.

<o> *
Specifies the field number within the frame.

Parameters:
<CondEnabler> ON | OFF
ON
Checking condition enabled

OFF
Checking condition disabled

*RST: OFF

Example:

`TRIGger:USBPd:FRAMe3:FLD1:ENABle ON`
Enables the checking condition for the Extended field of the Control frame.
`TRIGger:USBPd:FRENable "Data Frame", "Data Role" ON`
Enables the checking condition for the Data Role field of the Data frame.

TRIGger<m>:USBPd:BIT <Frame>, <Field>, <Bit>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:BIT <BitState>

Sets the bit state of a field that only consists of one bit.

Suffix:
<m> 1..3

<n> *

<o> *
Specifies the field number within the frame.

Parameters:
<BitState> ONE | ZERO | DC

ONE
1

ZERO
0

DC
"Don't care" (DC) = X

*RST: DC

TRIGger<m>:USBPd:DMAX <Frame>, <Field>, <Data>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMAX <DataMax>

Sets the end value of a data pattern range if `TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMAX` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

<n> *
Specifies the field number within the frame.

<o> *

Parameters:
<DataMax> Specifies the name of the generic trigger setting frame, see `BUS<m>:USBPd:RESult:FRAMe<n>:TYPE?` on page 2231.

TRIGger<m>:USBPd:DMIN <Frame>, <Field>, <Data>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMIN <DataMin>

Specifies the data pattern, or sets the start value of a data pattern range.

Suffix:

<m> 1..3

<n> *
Specifies the field number within the frame.

<o> *

Parameters:

<DataMin>

TRIGger<m>:USBPd:DOPerator <Frame>, <Field>, <Operator>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DOPerator <DataOperator>

Sets the operator for the data pattern in the selected field of the selected frame.

Suffix:

<m> 1..3

<n> *
Specifies the field number within the frame.

<o> *

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one endpoint value to be set using `TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMIN`.

INRange | OORange
In range, out of range. These conditions require a range of endpoint values to be set using `TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMIN` and `TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMAX`.

*RST: EQUal

TRIGger<m>:USBPd:IMAX <Frame>, <Field>, <Data>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMAX <IndexMax>

Sets the end value of an index range if `TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IOPerator` on page 2226 is set to `INRange` or `RANGe`.

Suffix:

<m> 1..3

<n> *
Specifies the field number within the frame.

<o> *

Parameters:

<IndexMax> Specifies the end value for the index range within the field. The index range, increment and *RST values depend on the field type.

TRIGger<m>:USBPd:IMIN <Frame>, <Field>, <Data>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMIN <IndexMin>

Specifies the index, or sets the start value of an index range.

Suffix:

<m> 1..3

<n> *
Specifies the field number within the frame.

<o> *

Parameters:

<IndexMin> Specifies the index value or index start value within the field. The index range, increment and *RST values depend on the field type.

TRIGger<m>:USBPd:IOPerator <Frame>, <Field>, <Operator>
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IOPerator <IndexOperator>

Sets the operator for the index in the selected field of the selected frame.

Suffix:

<m> 1..3

<n> *

<o> *

Parameters:

<IndexOperator> EQUal | INRange | RANGe

EQUal
This condition requires one endpoint value to be set using `TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMIN` and `TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMAX`.

*RST: INRange

24.17.23.3 Decode results

In all `BUS<m>:USBPd:RESult:FRAMe<n>` commands, the suffix <m> selects the serial bus and the suffix <n> selects the frame number in the decode table.

BUS<m>:USBPd:RESult:FCOunt?....2227
BUS<m>:USBPd:RESult:FRAMe<n>:DATA?....2227
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:FVAL?....2227
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STATus?....2228
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STARt?....2228
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STOP?....2229
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:TYPE?....2229
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:VAL?....2229
BUS<m>:USBPd:RESult:FRAMe<n>:INFO?....2230
BUS<m>:USBPd:RESult:FRAMe<n>:STARt?....2230
BUS<m>:USBPd:RESult:FRAMe<n>:STATe?....2230
BUS<m>:USBPd:RESult:FRAMe<n>:STOP?....2231
BUS<m>:USBPd:RESult:FRAMe<n>:TYPE?....2231

BUS<m>:USBPd:RESult:FCOunt?

Returns the number of decoded frames in the selected serial bus.

Suffix:
<m> 1..4

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:DATA?

Returns the data from the frame with the selected frame number (for the selected serial bus). This is the hex value of the first four bytes of data fields, prefixed by the total number of data fields (in square brackets), e.g. `[5]FF008041`.

Suffix:
<m> 1..4

<n> *

Return values:
<Data>

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:FVAL?

Returns the formatted value of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:
<FrameByteValue>

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STATus?

Returns the status of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<FieldState> OK | UNKNown | INComplete | PREamble | SOP | CRC | LENGth | FBFB

SOP
Start of packet warning

CRC
Cyclic redundancy check error

FBFB
4b5b error

*RST: OK

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STARt?

Returns the start time of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

<o> *

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STOP?

Returns the stop time of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

<o> *

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:TYPE?

Returns the type of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<FieldType>

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:VAL?

Returns the value of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:
<FrameByteValue>

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:INFO?

Returns information on the frame with the selected frame number (for the selected serial bus).

Suffix:
<m> 1..4

<n> *

Return values:
<Info>

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:STARt?

Returns the start time of the frame with the selected frame number (for the selected serial bus).

Suffix:
<m> 1..4

<n> *

Return values:
<Start>

Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:STATe?

Returns the status of the frame with the selected frame number (for the selected serial bus).

Suffix:
<m> 1..4

<n> *

Return values:

<State> OK | UNKNown | INComplete | PREamble | SOP | CRC | LENGth | FBFB

SOP
Start of packet warning

CRC
Cyclic redundancy check error

FBFB
4b5b error

*RST: OK

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:STOP?

Returns the stop time of the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

Return values:

<Stop>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:USBPd:RESult:FRAMe<n>:TYPE?

Returns the frame type for the selected serial bus and frame number.

Suffix:

<m> 1..4

<n> *

Return values:

<FrameType> STARt | RESet | CTRL | DATA | BIST | RQST | SINK | SRC | VEND | BATT | ALRT | XMSG | XSRC | XSTA | XGBC | XGBS | XBAC | XGMI | XMFI | XSRQ | XSRS | XFRQ | XFRS | TEST | LOWP | UNKNown

For a description of the frame types, see "Frame packet types" on page 862.

*RST: UNKNown

Usage: Query only

24.17.23.4 Search settings

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

There are two commands for each parameter, that you can use for defining the USBPD settings.

For example, to set the *Frame type =Control Frame > Field =MsgID >Data* value you can use one of the following commands:

- `SEARch:TRIGger:USBPD:FRAMe3:FLD3:DMIN 01100`
  Defines the parameter by using the index <m> for the frame number and <n> for the field number. For an overview, see Table 24-15.
- `SEARch:TRIGger:USBPD:DMIN "Control", "MsgID", 01100`
  Defines the parameter by using the frame and field name.

SEARch:TRIGger:USBPd:ERENable 2232
SEARch:TRIGger:USBPd:ERRor<m>:ENABle 2232
SEARch:TRIGger:USBPd:FRENable 2233
SEARch:TRIGger:USBPd:FRAMe<m>:ENABle 2233
SEARch:TRIGger:USBPd:FIENable 2233
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:ENABle 2233
SEARch:TRIGger:USBPd:BIT 2234
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:BIT 2234
SEARch:TRIGger:USBPd:DMAX 2234
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMAX 2234
SEARch:TRIGger:USBPd:DMIN 2235
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMIN 2235
SEARch:TRIGger:USBPd:DOPerator 2235
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DOPerator 2235
SEARch:TRIGger:USBPd:IMAX 2236
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMAX 2236
SEARch:TRIGger:USBPd:IMIN 2236
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMIN 2236
SEARch:TRIGger:USBPd:IOPerator 2237
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IOPerator 2237

SEARch:TRIGger:USBPd:ERENable <SearchName>,<ErrorName>, <Enabler>
SEARch:TRIGger:USBPd:ERRor<m>:ENABle <SearchName>,<Enable>
SEARch:TRIGger:USBPd:ERRor<m>:ENABle? <SearchName>

Enables or disables the checking condition for searching a specific error in the selected field of the selected frame.

Suffix:
<m> *
Specifies the error number.

Parameters:
<Enable> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

Example: `SEARch:TRIGger:USBPd:ERRor1:ENABle ON`
Enables searching for CRC Error.
`SEARch:TRIGger:USBPd:ERENable "Length Error",ON`
Enables searching for Length Error.

SEARch:TRIGger:USBPd:FRENable <SearchName>,<Frame>, <Enabler>
SEARch:TRIGger:USBPd:FRAMe<m>:ENABle <SearchName>,<Enable>
SEARch:TRIGger:USBPd:FRAMe<m>:ENABle? <SearchName>
Suffix:
<m> *
Specifies the frame number.

Parameters:
<Enable> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

Example: `SEARch:TRIGger:USBPd:FRAMe1:ENABle ON`
Enables the checking condition for the Test frame.
`SEARch:TRIGger:USBPd:FRENable "Control Frame" ON`
Enables the checking condition for the Control frame.

SEARch:TRIGger:USBPd:FIENable <SearchName>,<Frame>, <Field>, <Enabler>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:ENABle
<SearchName>,<CondEnabler>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:ENABle? <SearchName>

Enables or disables the checking condition for searching a specific data pattern in the selected field of the selected frame.

Suffix:
<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:
<CondEnabler> ON | OFF

ON
Checking condition enabled

OFF
Checking condition disabled

*RST: OFF

Parameters for setting and query:
<SearchName> String with the name of the search.

Example: `SEARch:TRIGger:USBPd:FRAMe3:FLD1:ENABle ON`
Enables searching for the Extended field of the Control frame.
`SEARch:TRIGger:USBPd:FRENable "Data Frame", "Data Role" ON`
Enables searching for the Data Role field of the Data frame.

SEARch:TRIGger:USBPd:BIT <SearchName>,<Frame>, <Field>, <Bit>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:BIT <SearchName>,<BitState>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:BIT? <SearchName>

Sets the bit state of a field to be searched that only consists of one bit.

Suffix:
<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:
<BitState> ONE | ZERO | DC

ONE
1

ZERO
0

DC
"Don't care" = X

*RST: DC

Parameters for setting and query:
<SearchName> String with the name of the search.

SEARch:TRIGger:USBPd:DMAX <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMAX <SearchName>,<DataMax>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMAX? <SearchName>

Sets the end value of a data pattern range if `SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DOPerator` is set to `INRange` or `OORange`.

Suffix:
<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:
<DataMax> Specifies the name of the generic trigger setting frame, see `SEARch:RESult:USBPd:FRAMe<m>:TYPE?` on page 2242.

Parameters for setting and query:
<SearchName> String with the name of the search.

SEARch:TRIGger:USBPd:DMIN <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMIN <SearchName>,<DataMin>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMIN? <SearchName>

Specifies the data pattern to be searched, or sets the start value of a data pattern range to be searched.

Suffix:
<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:
<DataMin> Specifies the name of the generic trigger setting frame, see `SEARch:RESult:USBPd:FRAMe<m>:TYPE?` on page 2242.

Parameters for setting and query:
<SearchName> String with the name of the search.

SEARch:TRIGger:USBPd:DOPerator <SearchName>,<Frame>, <Field>, <Operator>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DOPerator
<SearchName>,<DataOperator>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DOPerator? <SearchName>

Sets the operator to set a specific data pattern to be searched in the selected field of the selected frame.

Suffix:
<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:
<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one endpoint value to be set using `SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMIN`.

INRange | OORange
In range, out of range. These conditions require a range of endpoint values to be set using `SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMIN` and `SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName> String with the name of the search.

SEARch:TRIGger:USBPd:IMAX <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMAX <SearchName>,<IndexMax>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMAX? <SearchName>

Sets the end value of an index range if `SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IOPerator` is set to `INRange` or `RANGe`.

Suffix:
<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:
<IndexMax> Specifies the end value for the index range within the field. The index range, increment and *RST values depend on the field type.

Parameters for setting and query:
<SearchName> String with the name of the search.

SEARch:TRIGger:USBPd:IMIN <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMIN <SearchName>,<IndexMin>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMIN? <SearchName>

Specifies the index at which the data is to be searched, or sets the start value of an index range in which the data is to be searched.

Suffix:
<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:

<IndexMin> Specifies the index value or the start value of an index range within the field.
The index range, increment and *RST values depend on the field type.

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:USBPd:IOPerator <SearchName>,<Frame>, <Field>, <Operator>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IOPerator
<SearchName>,<IndexOperator>
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IOPerator? <SearchName>

Sets the operator for the index for searching in the selected field of the selected frame.

Suffix:

<m> *
Specifies the frame.

<n> *
Specifies the field number within the frame.

Parameters:

<IndexOperator> EQUal | INRange | RANGe

EQUal
This condition requires one endpoint value to be set using `SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMIN` and `SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMAX`.

*RST: INRange

Parameters for setting and query:

<SearchName> String with the name of the search.

24.17.23.5 Search results

In all `SEARch:RESult:USBPd:FRAMe<m>` commands, the suffix <m> selects the frame number in the list of search results.

SEARch:RESult:USBPd:FCOunt?..... 2238
SEARch:RESult:USBPd:FRAMe<m>:DATA?..... 2238
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:FVAL?..... 2238
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STATus?..... 2239
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STARt?..... 2239
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STOP?..... 2239
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:TYPE?..... 2240
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:VAL?..... 2240
SEARch:RESult:USBPd:FRAMe<m>:INFO?..... 2241

SEARch:RESult:USBPd:FRAMe<m>:STARt?.....2241
SEARch:RESult:USBPd:FRAMe<m>:STATe?.....2241
SEARch:RESult:USBPd:FRAMe<m>:STOP?.....2242
SEARch:RESult:USBPd:FRAMe<m>:TYPE?.....2242

SEARch:RESult:USBPd:FCOunt? <SearchName>

Returns the number of decoded frames in the search result.

Query parameters:
<SearchName>

Return values:

<FrameCount>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:DATA? <SearchName>

Searches for data from the frame with the selected frame number (for the selected serial bus). This is the hex value of the first four bytes of data fields, prefixed by the total number of data fields (in square brackets), e.g. `[5]FF008041`.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:FVAL? <SearchName>

Returns the formatted value of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *
Selects the field number.

Query parameters:
<SearchName>

Return values:
<ValueFormatted>

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STATus? <SearchName>

Returns the field status of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *
Selects the field number.

Query parameters:
<SearchName>

Return values:
<FieldState> OK | UNKNown | INComplete | PREamble | SOP | CRC | LENGth | FBFB

SOP
Start of packet warning

CRC
Cyclic redundancy check error

FBFB
4b5b error

*RST: OK

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STARt? <SearchName>

Returns the start time of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STOP? <SearchName>

Returns the stop time of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:TYPE? <SearchName>

Returns the field name of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *
Selects the field number.

Query parameters:
<SearchName>

Return values:
<FieldType>

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:VAL? <SearchName>

Returns the value of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *
Selects the field number.

Query parameters:
<SearchName>

Return values:
<ByteValue>

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:INFO? <SearchName>

Returns specific frame information details of the selected frame in the search result. This information is also shown in the "Info" column of the decode results table.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Info>

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:STARt? <SearchName>

Returns the start time of the frame with the selected frame number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Start>

Range: -100E+24 to 100E+24
Increment: 100E-12
Default unit: s

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:STATe? <SearchName>

Returns the status of the frame with the selected frame number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State>

OK | UNKNown | INComplete | PREamble | SOP | CRC | LENGth | FBFB

SOP
Start of packet warning

CRC
Cyclic redundancy check error

FBFB
4b5b error

*RST: OK

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:STOP? <SearchName>

Returns the stop time of the frame with the selected frame number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
Default unit: s

Usage: Query only

SEARch:RESult:USBPd:FRAMe<m>:TYPE? <SearchName>

Returns the frame type for the selected frame number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameType> STARt | RESet | CTRL | DATA | BIST | RQST | SINK | SRC | VEND | BATT | ALRT | XMSG | XSRC | XSTA | XGBC | XGBS | XBAC | XGMI | XMFI | XSRQ | XSRS | XFRQ | XFRS | TEST | LOWP | UNKNown

For a description of the frame types, see "Frame packet types" on page 862.

*RST: UNKNown

Usage: Query only

24.17.24 D-PHY (Option R&S RTO6-K580)

- D-PHY configuration 2243
- D-PHY trigger 2248
- D-PHY decode results 2255
- D-PHY search settings 2261
- D-PHY search results 2269

24.17.24.1 D-PHY configuration

BUS<m>:DPHY:CONFig....2243
BUS<m>:DPHY:CP:PROBe....2244
BUS<m>:DPHY:CP:SOURce....2244
BUS<m>:DPHY:DRATe....2244
BUS<m>:DPHY:DSPData....2245
BUS<m>:DPHY:THCoupling....2245
BUS<m>:DPHY:THPReset....2245
BUS<m>:DPHY:VARiant....2245
BUS<m>:DPHY:DNZero:LPOWer:THUPper....2246
BUS<m>:DPHY:DPZero:LPOWer:THUPper....2246
BUS<m>:DPHY:DNZero:LPOWer:THLower....2246
BUS<m>:DPHY:DPZero:LPOWer:THLower....2246
BUS<m>:DPHY:DPZero:SOURce....2246
BUS<m>:DPHY:DNZero:SOURce....2246
BUS<m>:DPHY:DPONe:SOURce....2246
BUS<m>:DPHY:DPTWo:SOURce....2246
BUS<m>:DPHY:DPTHree:SOURce....2246
BUS<m>:DPHY:DNZero:PROBe....2247
BUS<m>:DPHY:DPZero:PROBe....2247
BUS<m>:DPHY:DPONe:PROBe....2247
BUS<m>:DPHY:DPTWo:PROBe....2247
BUS<m>:DPHY:DPTHree:PROBe....2247
BUS<m>:DPHY:CP:HSPeed:HYSTeresis....2247
BUS<m>:DPHY:DPZero:HSPeed:HYSTeresis....2247
BUS<m>:DPHY:DPONe:HSPeed:HYSTeresis....2247
BUS<m>:DPHY:DPTWo:HSPeed:HYSTeresis....2247
BUS<m>:DPHY:DPTHree:HSPeed:HYSTeresis....2247
BUS<m>:DPHY:CP:HSPeed:THReshold....2247
BUS<m>:DPHY:DPZero:HSPeed:THReshold....2247
BUS<m>:DPHY:DPONe:HSPeed:THReshold....2247
BUS<m>:DPHY:DPTWo:HSPeed:THReshold....2247
BUS<m>:DPHY:DPTHree:HSPeed:THReshold....2247
BUS<m>:DPHY:SMLanes....2248

BUS<m>:DPHY:CONFig <Configuration>

Sets the number of data lanes and if the clock lane and low power are available.

Suffix:

<m> 1..4

Parameters:

<Configuration> D1CN | D1CL | D2CN | D2CL | D3CN | D3NL | D4NN

`D1CN`: 1 data lane, clock lane
`D1CL`: 1 data lane, clock lane, low power
`D2CN`: 2 data lanes, clock lane
`D2CL`: 2 data lanes, clock lane, low power
`D3CN`: 3 data lanes, clock lane

`D3CL`: 3 data lanes, clock lane, low power
`D4NN`: 4 data lanes and no clock

*RST: D4NN

BUS<m>:DPHY:CP:PROBe <ProbeCP>

Selects the type of probe used for the clock lane.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ProbeCP> DIFFerential | SINGle

*RST: SINGle

BUS<m>:DPHY:CP:SOURce <ClockSource>

Selects the source of the clock lane.

Usually, the source is one of the analog channels. Reference and math waveforms are available as source if the trigger source is one of the analog channels but not the serial bus.

Alternatively, digital channels can be used if MSO option R&S RTO6-B1 is installed. Digital and analog channels cannot be used at the same time.

For triggering on a serial bus, analog or digital channel sources are required.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ClockSource> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: NONE

BUS<m>:DPHY:DRATe <DataRate>

Sets a data rate.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<DataRate> Range: 1000000 to 2500000000
Increment: 1000000
*RST: 800000000
Default unit: bps

BUS<m>:DPHY:DSPData <DispDatVect>

Enables the display of the data vector, which means that the whole frame will be decoded.

When the display vector is disabled only the packet header will be decoded and not the actual data payload of the frame. This will speed up the decoding.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<DispDatVect> ON | OFF
*RST: OFF

BUS<m>:DPHY:THCoupling <ThresCpl>

Enables coupling, i.e. the same threshold and hysteresis value is used for all lanes.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ThresCpl> ON | OFF
*RST: ON

BUS<m>:DPHY:THPReset <ThresholdPreset>

Prests the threshold and hysteresis values of the high speed data lanes. A preset sets the low power threshold to 1.20V and high speed threshold to 200 mV.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ThresholdPreset> LPHS | MANual
*RST: LPHS

BUS<m>:DPHY:VARiant <ProtSel>

Selects the protocol running on the interface.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ProtSel> DSI | CSI2
*RST: CSI2

BUS<m>:DPHY:DNZero:LPOWer:THUPper <ThresLPDN0High>
BUS<m>:DPHY:DPZero:LPOWer:THUPper <ThresLPDP0High>

Sets the upper threshold value for the respective lane used for the low power mode.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ThresLPDP0High> Range: -1.5 to 1.5
Increment: 0.01
*RST: 0.88
Default unit: V

BUS<m>:DPHY:DNZero:LPOWer:THLower <ThresLPDN0Low>
BUS<m>:DPHY:DPZero:LPOWer:THLower <ThresLPDP0Low>

Sets the lower threshold value for the respective lane used for the low power mode.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ThresLPDP0Low> Range: -1.5 to 1.5
Increment: 0.01
*RST: 0.55
Default unit: V

BUS<m>:DPHY:DPZero:SOURce <DataSource>

Selects the source of the DP0 data line.

Suffix:
<m> 1..4

Parameters:
<DataSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4
*RST: C1W1

BUS<m>:DPHY:DNZero:SOURce <DataSource>
BUS<m>:DPHY:DPONe:SOURce <DataSource>
BUS<m>:DPHY:DPTWo:SOURce <DataSource>
BUS<m>:DPHY:DPTHree:SOURce <DataSource>

Selects the source of the corresponding data line.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:

<DataSource> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

*RST: DNZero: NONE, DPONe: C2W1, DPTWo: C3W1, DPTHree: C4W1

BUS<m>:DPHY:DNZero:PROBe <ProbeDN0>
BUS<m>:DPHY:DPZero:PROBe <ProbeDP0>
BUS<m>:DPHY:DPONe:PROBe <ProbeDP1>
BUS<m>:DPHY:DPTWo:PROBe <ProbeDP2>
BUS<m>:DPHY:DPTHree:PROBe <ProbeDP3>

Selects the type of probe used for the corresponding data lane.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<ProbeDP3> DIFFerential | SINGle

*RST: SINGle

BUS<m>:DPHY:CP:HSPeed:HYSTeresis <HysteresisHSCP>
BUS<m>:DPHY:DPZero:HSPeed:HYSTeresis <HysteresisHSDP0>
BUS<m>:DPHY:DPONe:HSPeed:HYSTeresis <HysteresisHSDP1>
BUS<m>:DPHY:DPTWo:HSPeed:HYSTeresis <HysteresisHSDP2>
BUS<m>:DPHY:DPTHree:HSPeed:HYSTeresis <HysteresisHSDP3>

Sets a value for the hysteresis of the respective lane.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<HysteresisHSDP3> Range: 0 to 0.5
Increment: 0.01
*RST: 0.2
Default unit: V

BUS<m>:DPHY:CP:HSPeed:THReshold <ThresholdHSCP>
BUS<m>:DPHY:DPZero:HSPeed:THReshold <ThresholdHSDP0>
BUS<m>:DPHY:DPONe:HSPeed:THReshold <ThresholdHSDP1>
BUS<m>:DPHY:DPTWo:HSPeed:THReshold <ThresholdHSDP2>
BUS<m>:DPHY:DPTHree:HSPeed:THReshold <ThresholdHSDP3>

Sets the threshold value for the digitization of the respective high speed data line.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<ThresholdHSDP3>	Range:	-1 to 1
	Increment:	0.01
	*RST:	0.2
	Default unit:	V

BUS<m>:DPHY:SMLanes <Active>

Enables the display of the results in multiple lanes.

Suffix:

<m>	1..4

Parameters:

<Active>	ON | OFF	
	*RST:	ON

24.17.24.2 D-PHY trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:DPHY:TYPE 2249
TRIGger<m>:DPHY:DATA:CONDition 2249
TRIGger<m>:DPHY:DATA:MAX 2250
TRIGger<m>:DPHY:DATA:MIN 2250
TRIGger<m>:DPHY:DIDX:CONDition 2250
TRIGger<m>:DPHY:DIDX:MAX 2250
TRIGger<m>:DPHY:DIDX:MIN 2251
TRIGger<m>:DPHY:DTYPe:CONDition 2251
TRIGger<m>:DPHY:DTYPe:MAX 2251
TRIGger<m>:DPHY:DTYPe:MIN 2251
TRIGger<m>:DPHY:ESCMode:CONDition 2252
TRIGger<m>:DPHY:ESCMode:MAX 2252
TRIGger<m>:DPHY:ESCMode:MIN 2252
TRIGger<m>:DPHY:ESDTa:CONDition 2252
TRIGger<m>:DPHY:ESDTa:MAX 2253
TRIGger<m>:DPHY:ESDTa:MIN 2253
TRIGger<m>:DPHY:ESINdex:CONDition 2253
TRIGger<m>:DPHY:ESINdex:MAX 2254
TRIGger<m>:DPHY:ESINdex:MIN 2254
TRIGger<m>:DPHY:HSVC 2254

TRIGger<m>:DPHY:WCOunt:CONDition....2254
TRIGger<m>:DPHY:WCOunt:MAX.... 2254
TRIGger<m>:DPHY:WCOunt:MIN.... 2255

TRIGger<m>:DPHY:TYPE <Type>

Sets the type of frame to be triggered on.

Parameters:

<Type> HS_SOP | HS_EOP | HS_PH | HS_DATA | LP_ESC | LP_TURN | LP_HSRQ

HS_SOP
High speed start of packet

HS_EOP
High speed end of packet

HS_PH
High speed packet header

HS_DATA
High speed data

LP_ESC
Low power escape mode frame

LP_TURN
Low power lane turnaround

LP_HSRQ
Low power high speed request

*RST: HS_SOP

TRIGger<m>:DPHY:DATA:CONDition <Format>

Set the condition for the data value. You can define an exact value or a value range.

Parameters:

<Format> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:DPHY:DATA:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:DPHY:DATA:MIN` and `TRIGger<m>:DPHY:DATA:MAX`.

*RST: EQUal

TRIGger<m>:DPHY:DATA:MAX <DataMax>

Sets the the end value of a data type range if `TRIGger<m>:DPHY:DATA:CONDition` is set to `INRange` or `OORange`.

Parameters:

<DataMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:DPHY:DATA:MIN <DataMin>

Specifies a data bit pattern, or sets the the start value of a pattern range.

Parameters:

<DataMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:DPHY:DIDX:CONDition <Format>

Sets the condition for the data index. You can define an exact value or a value range.

Parameters:

<Format> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `TRIGger<m>:DPHY:DIDX:MIN` command.

INRange = RANGe
In range: Set the minimum and maximum value using `TRIGger<m>:DPHY:DIDX:MIN` and `TRIGger<m>:DPHY:DIDX:MAX`.

*RST: INRange

TRIGger<m>:DPHY:DIDX:MAX <DXSymbol>

Sets the the end value of a data index range if `TRIGger<m>:DPHY:DIDX:CONDition` is set to `INRange` or `RANge`.

Parameters:

<DXSymbol> Range: 0 to 65535
Increment: 1
*RST: 0

TRIGger<m>:DPHY:DIDX:MIN <SymbolType>

Specifies a data index minimum, or sets the the start value of a range.

Parameters:

<SymbolType>

Range: 0 to 65535
Increment: 1
*RST: 0

TRIGger<m>:DPHY:DTYPe:CONDition <Format>

Set the condition for the data type. You can define an exact value or a value range

Parameters:

<Format>

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `TRIGger<m>:DPHY:DTYPe:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:DPHY:DTYPe:MIN` and `TRIGger<m>:DPHY:DTYPe:MAX`.

*RST: EQUal

TRIGger<m>:DPHY:DTYPe:MAX <HSDataTypeMax>

Sets the the end value of a data type range if `TRIGger<m>:DPHY:DTYPe:CONDition` is set to `INRange` or `OORange`.

Parameters:

<HSDataTypeMax>

Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:DPHY:DTYPe:MIN <HSDataTypeMin>

Specifies a data type pattern, or sets the the start value of a pattern range.

Parameters:

<HSDataTypeMin>

Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:DPHY:ESCMode:CONDition <Format>

Set the condition for the escape mode. You can define an exact value or a value range

Parameters:

<Format> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `TRIGger<m>:DPHY:ESCMode:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:DPHY:ESCMode:MIN` and `TRIGger<m>:DPHY:ESCMode:MAX`.

*RST: EQUal

TRIGger<m>:DPHY:ESCMode:MAX <LPEscapeModeMax>

Sets the the end value of an escape mode range if `TRIGger<m>:DPHY:ESCMode:CONDition` is set to `INRange` or `OORange`.

Parameters:

<LPEscapeModeMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:DPHY:ESCMode:MIN <LPEscapeModeMin>

Specifies a escape mode bit pattern, or sets the the start value of a pattern range.

Parameters:

<LPEscapeModeMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:DPHY:ESDTa:CONDition <Format>

Set the condition for the escape mode data value. You can define an exact value or a value range.

Parameters:

<Format> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:DPHY:ESDTa:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:DPHY:ESDTa:MIN` and `TRIGger<m>:DPHY:ESDTa:MAX`.

OFF | ANY
Any pattern is detected.

*RST: EQUal

TRIGger<m>:DPHY:ESDTa:MAX <LPEscapeDataMax>

Sets the the end value of a escape mode data type range if `TRIGger<m>:DPHY:ESDTa:CONDition` is set to `INRange` or `OORange`.

Parameters:
<LPEscapeDataMax>

TRIGger<m>:DPHY:ESDTa:MIN <LPEscapeDataMin>

Specifies an escape mode data bit pattern, or sets the the start value of a pattern range.

Parameters:
<LPEscpMdDatMin>

TRIGger<m>:DPHY:ESINdex:CONDition <Format>

Sets the condition for the escape mode data index. You can define an exact value or a value range.

Parameters:
<Format> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `TRIGger<m>:DPHY:ESINdex:MIN` command.

INRange = RANGe
In range: Set the minimum and maximum value using `TRIGger<m>:DPHY:ESINdex:MIN` and `TRIGger<m>:DPHY:ESINdex:MAX`.

*RST: INRange

TRIGger<m>:DPHY:ESINdex:MAX <DXSymbol>

Sets the the end value of a data index range if `TRIGger<m>:DPHY:ESINdex:CONDition` is set to `INRange` or `RANge`.

Parameters:

<DXSymbol> Range: 0 to 255
Increment: 1
*RST: 0

TRIGger<m>:DPHY:ESINdex:MIN <SymbolType>

Specifies an escape mode data index minimum, or sets the the start value of a range.

Parameters:

<SymbolType> Range: 0 to 255
Increment: 1
*RST: 0

TRIGger<m>:DPHY:HSVC <HSVC>

Sets the virtual channel to be triggered on.

Parameters:

<HSVC>

TRIGger<m>:DPHY:WCOunt:CONDition <Format>

Set the condition for the word count. You can define an exact value or a value range.

Parameters:

<Format> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `TRIGger<m>:DPHY:WCOunt:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:DPHY:WCOunt:MIN` and `TRIGger<m>:DPHY:WCOunt:MAX`.

*RST: EQUal

TRIGger<m>:DPHY:WCOunt:MAX <HSWordCountMax>

Sets the the end value of a data type range if `TRIGger<m>:DPHY:WCOunt:CONDition` is set to `INRange` or `OORange`.

Parameters:

<HSWordCountMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:DPHY:WCOunt:MIN <HSWordCountMin>

Specifies a word bit pattern, or sets the the start value of a pattern range.

Parameters:

<HSWordCountMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

24.17.24.3 D-PHY decode results

To show the results on the screen, use the following commands:

- `BUS<m>:RESult` on page 1671

BUS<m>:DPHY:RESult:FCOunt? 2255
BUS<m>:DPHY:RESult:FRAMe<n>:CS? 2255
BUS<m>:DPHY:RESult:FRAMe<n>:DATA? 2256
BUS<m>:DPHY:RESult:FRAMe<n>:DTName? 2256
BUS<m>:DPHY:RESult:FRAMe<n>:DTYPe? 2257
BUS<m>:DPHY:RESult:FRAMe<n>:ECC? 2257
BUS<m>:DPHY:RESult:FRAMe<n>:PACKet<o>:IDX? 2257
BUS<m>:DPHY:RESult:FRAMe<n>:PACKet<o>:VALue? 2258
BUS<m>:DPHY:RESult:FRAMe<n>:STARt? 2258
BUS<m>:DPHY:RESult:FRAMe<n>:STATe? 2258
BUS<m>:DPHY:RESult:FRAMe<n>:STOP? 2259
BUS<m>:DPHY:RESult:FRAMe<n>:TYPE? 2259
BUS<m>:DPHY:RESult:FRAMe<n>:NUMPackets? 2260
BUS<m>:DPHY:RESult:FRAMe<n>:VCHannel? 2260

BUS<m>:DPHY:RESult:FCOunt?

Returns the number of decoded frames for the selected serial bus.

Suffix:

<m> 1..4

Return values:

<Count>
Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:DPHY:RESult:FRAMe<n>:CS?

Returns the checksum of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<CS> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:DPHY:RESult:FRAMe<n>:DATA?

Returns the data or word count value.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<WordCountOrData> 16 bit integer value
Range: 0 to 65535
Increment: 1
*RST: 0

Example:

```
BUS:DPHY:RESult:FRAMe2:DATA?
<-- 13245
```

Usage: Query only

BUS<m>:DPHY:RESult:FRAMe<n>:DTName?

Returns the data type name for the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<DataTypeName>

Usage: Query only

BUS<m>:DPHY:RESult:FRAMe<n>:DTYPe?

Returns the data type for the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<DataType> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:DPHY:RESult:FRAMe<n>:ECC?

Returns the error correction code (ECC) of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<ECC> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:DPHY:RESult:FRAMe<n>:PACKet<o>:IDX?

Returns the index of the data payload.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the packet number.

Return values:

<PacketIndex> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:DPHY:RESult:FRAMe<n>:PACKet<o>:VALue?

Returns the byte value of the data payload.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the packet number.

Return values:

<PacketValue> Range: 0 to 9223372036854775808
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:DPHY:RESult:FRAMe<n>:STARt?

Returns the start time of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:DPHY:RESult:FRAMe<n>:STATe?

Returns the overall state of the frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<State> OK | CS | ECC | ECCWarning | LENGth | FORMat | UNKNOWN_FRAME | INComplete

OK
The frame is valid.

LENGth
Length error.

UNKNOWN_FRAME
Unknown frame type

INComplete
The sequence is not completely contained in the acquisition

*RST: OK

Usage: Query only

BUS<m>:DPHY:RESult:FRAMe<n>:STOP?

Returns the end time of the specified frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:DPHY:RESult:FRAMe<n>:TYPE?

Returns the type of frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameType> HS_SP | HS_LP | HS_INCOMPLETE | LP_HSREQ | LP_TA | LP_ESC

HS_SP
High speed short packet frame

HS_LP
High speed long packet frame

HS_INCOMPLETE
Incomplete high speed frame

LP_HSREQ
High speed request frame

LP_TA
Low power lane turnaround frame

LP_ESC
Low power escape mode frame

*RST: HS_SP

Usage: Query only

BUS<m>:DPHY:RESult:FRAMe<n>:NUMPackets?

Returns the number of packets in the indicated frame.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<NumPackets> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:DPHY:RESult:FRAMe<n>:VCHannel?

Returns the number of virtual channels.

Suffix:
<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<VirtualChannel> Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

24.17.24.4 D-PHY search settings

SEARch:TRIGger:DPHY:TYPE....2261
SEARch:TRIGger:DPHY:DATA:CONDition....2262
SEARch:TRIGger:DPHY:DATA:MAX....2262
SEARch:TRIGger:DPHY:DATA:MIN....2262
SEARch:TRIGger:DPHY:DIDX:CONDition....2263
SEARch:TRIGger:DPHY:DIDX:MAX....2263
SEARch:TRIGger:DPHY:DIDX:MIN....2263
SEARch:TRIGger:DPHY:DTYPe:CONDition....2264
SEARch:TRIGger:DPHY:DTYPe:MAX....2264
SEARch:TRIGger:DPHY:DTYPe:MIN....2264
SEARch:TRIGger:DPHY:ESCMode:CONDition....2265
SEARch:TRIGger:DPHY:ESCMode:MAX....2265
SEARch:TRIGger:DPHY:ESCMode:MIN....2265
SEARch:TRIGger:DPHY:ESDTa:CONDition....2266
SEARch:TRIGger:DPHY:ESDTa:MAX....2266
SEARch:TRIGger:DPHY:ESDTa:MIN....2266
SEARch:TRIGger:DPHY:ESINdex:CONDition....2267
SEARch:TRIGger:DPHY:ESINdex:MAX....2267
SEARch:TRIGger:DPHY:ESINdex:MIN....2267
SEARch:TRIGger:DPHY:HSVC....2268
SEARch:TRIGger:DPHY:WCOunt:CONDition....2268
SEARch:TRIGger:DPHY:WCOunt:MAX....2268
SEARch:TRIGger:DPHY:WCOunt:MIN....2269

SEARch:TRIGger:DPHY:TYPE <SearchName>,<Type>
SEARch:TRIGger:DPHY:TYPE? <SearchName>

Sets the type of frame to be searched for.

Parameters:

<Type> HS_SOP | HS_EOP | HS_PH | HS_DATA | LP_ESC | LP_TURN | LP_HSRQ

HS_SOP
High speed start of packet

HS_EOP
High speed end of packet

HS_PH
High speed packet header

HS_DATA
High speed data

LP_ESC
Low power escape mode frame

LP_TURN
Low power lane turnaround

LP_HSRQ
Low power high speed request

*RST: HS_SOP

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:DATA:CONDition <SearchName>,<Format>
SEARch:TRIGger:DPHY:DATA:CONDition? <SearchName>

Set the condition for the data value. You can define an exact value or a value range.

Parameters:
<Format> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:DPHY:DATA:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:DPHY:DATA:MIN` and `SEARch:TRIGger:DPHY:DATA:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:DATA:MAX <SearchName>,<DataMax>
SEARch:TRIGger:DPHY:DATA:MAX? <SearchName>

Sets the the end value of a data type range if `SEARch:TRIGger:DPHY:DATA:CONDition` is set to `INRange` or `OORange`.

Parameters:
<DataMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:DATA:MIN <SearchName>,<DataMin>
SEARch:TRIGger:DPHY:DATA:MIN? <SearchName>

Specifies a data bit pattern, or sets the the start value of a pattern range.

Parameters:
<DataMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:DIDX:CONDition <SearchName>,<Format>
SEARch:TRIGger:DPHY:DIDX:CONDition? <SearchName>

Sets the condition for the data index. You can define an exact value or a value range.

Parameters:
<Format> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `SEARch:TRIGger:DPHY:DIDX:MIN` command.

INRange = RANGe
In range: Set the minimum and maximum value using `SEARch:TRIGger:DPHY:DIDX:MIN` and `SEARch:TRIGger:DPHY:DIDX:MAX`.

*RST: INRange

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:DIDX:MAX <SearchName>,<DXSymbol>
SEARch:TRIGger:DPHY:DIDX:MAX? <SearchName>

Sets the the end value of a data index range if `SEARch:TRIGger:DPHY:DIDX:CONDition` is set to `INRange` or `RANge`.

Parameters:
<DXSymbol> Range: 0 to 65535
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:DIDX:MIN <SearchName>,<SymbolType>
SEARch:TRIGger:DPHY:DIDX:MIN? <SearchName>

Specifies a data index minimum, or sets the the start value of a range.

Parameters:
<SymbolType> Range: 0 to 65535
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:DTYPe:CONDition <SearchName>,<Format>
SEARch:TRIGger:DPHY:DTYPe:CONDition? <SearchName>

Set the condition for the data type. You can define an exact value or a value range

Parameters:

<Format> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `SEARch:TRIGger:DPHY:DTYPe:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:DPHY:DATA:MIN` and `SEARch:TRIGger:DPHY:DATA:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:DTYPe:MAX <SearchName>,<HSDataTypeMax>
SEARch:TRIGger:DPHY:DTYPe:MAX? <SearchName>

Sets the the end value of a data type range if `SEARch:TRIGger:DPHY:DTYPe:CONDition` is set to `INRange` or `OORange`.

Parameters:

<HSDataTypeMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:DTYPe:MIN <SearchName>,<HSDataTypeMin>
SEARch:TRIGger:DPHY:DTYPe:MIN? <SearchName>

Specifies a data type pattern, or sets the the start value of a pattern range.

Parameters:

<HSDataTypeMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:ESCMode:CONDition <SearchName>,<Format>
SEARch:TRIGger:DPHY:ESCMode:CONDition? <SearchName>

Set the condition for the escape mode. You can define an exact value or a value range

Parameters:
<Format> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `SEARch:TRIGger:DPHY:ESCMode:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:DPHY:ESCMode:MIN` and `SEARch:TRIGger:DPHY:ESCMode:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:ESCMode:MAX <SearchName>,<LPEscapeModeMax>
SEARch:TRIGger:DPHY:ESCMode:MAX? <SearchName>

Sets the the end value of an escape mode range if `SEARch:TRIGger:DPHY:ESCMode:CONDition` is set to `INRange` or `OORange`.

Parameters:
<LPEscapeModeMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:ESCMode:MIN <SearchName>,<LPEscapeModeMin>
SEARch:TRIGger:DPHY:ESCMode:MIN? <SearchName>

Specifies a escape mode bit pattern, or sets the the start value of a pattern range.

Parameters:
<LPEscapeModeMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:ESDTa:CONDition <SearchName>,<Format>
SEARch:TRIGger:DPHY:ESDTa:CONDition? <SearchName>

Set the condition for the escape mode data value. You can define an exact value or a value range.

Parameters:
<Format> OFF | ANY | EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:DPHY:ESDTa:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:DPHY:ESDTa:MIN` and `SEARch:TRIGger:DPHY:ESDTa:MAX`.

OFF | ANY
Any pattern is detected.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:ESDTa:MAX <SearchName>,<LPEscapeDataMax>
SEARch:TRIGger:DPHY:ESDTa:MAX? <SearchName>

Sets the the end value of a escape mode data type range if `SEARch:TRIGger:DPHY:ESDTa:CONDition` is set to `INRange` or `OORange`.

Parameters:
<LPEscapeDataMax>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:ESDTa:MIN <SearchName>,<LPEscapeDataMin>
SEARch:TRIGger:DPHY:ESDTa:MIN? <SearchName>

Specifies an escape mode data bit pattern, or sets the the start value of a pattern range.

Parameters:
<LPEscapeDataMin>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:ESINdex:CONDition <SearchName>,<Format>
SEARch:TRIGger:DPHY:ESINdex:CONDition? <SearchName>

Sets the condition for the escape mode data index. You can define an exact value or a value range.

Parameters:
<Format> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `SEARch:TRIGger:DPHY:ESINdex:MIN` command.

INRange = RANGe
In range: Set the minimum and maximum value using `SEARch:TRIGger:DPHY:ESINdex:MIN` and `SEARch:TRIGger:DPHY:ESINdex:MAX`.

*RST: INRange

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:ESINdex:MAX <SearchName>,<DXSymbol>
SEARch:TRIGger:DPHY:ESINdex:MAX? <SearchName>

Sets the the end value of a data index range if `SEARch:TRIGger:DPHY:ESINdex:CONDition` is set to `INRange` or `RANge`.

Parameters:
<DXSymbol> Range: 0 to 255
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:ESINdex:MIN <SearchName>,<SymbolType>
SEARch:TRIGger:DPHY:ESINdex:MIN? <SearchName>

Specifies an escape mode data index minimum, or sets the the start value of a range.

Parameters:
<SymbolType> Range: 0 to 255
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:HSVC <SearchName>,<HSVC>
SEARch:TRIGger:DPHY:HSVC? <SearchName>

Sets the virtual channel to be searched for.

Parameters:
<HSVC>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:WCOunt:CONDition <SearchName>,<Format>
SEARch:TRIGger:DPHY:WCOunt:CONDition? <SearchName>

Set the condition for the word count. You can define an exact value or a value range.

Parameters:
<Format> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `SEARch:TRIGger:DPHY:WCOunt:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:DPHY:WCOunt:MIN` and `SEARch:TRIGger:DPHY:WCOunt:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:WCOunt:MAX <SearchName>,<HSWordCountMax>
SEARch:TRIGger:DPHY:WCOunt:MAX? <SearchName>

Sets the the end value of a data type range if `SEARch:TRIGger:DPHY:WCOunt:CONDition` is set to `INRange` or `OORange`.

Parameters:
<HSWordCountMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DPHY:WCOunt:MIN <SearchName>,<HSWordCountMin>
SEARch:TRIGger:DPHY:WCOunt:MIN? <SearchName>

Specifies a word bit pattern, or sets the the start value of a pattern range.

Parameters:

<HSWordCountMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

24.17.24.5 D-PHY search results

In all `SEARch:RESult:DPHY:FRAMe<m>` commands, the suffix <m> selects the frame number in the list of search results.

SEARch:RESult:DPHY:FCOunt? 2269
SEARch:RESult:DPHY:FRAMe<m>:CS? 2269
SEARch:RESult:DPHY:FRAMe<m>:DATA? 2270
SEARch:RESult:DPHY:FRAMe<m>:DTName? 2270
SEARch:RESult:DPHY:FRAMe<m>:DTYPe? 2270
SEARch:RESult:DPHY:FRAMe<m>:ECC? 2271
SEARch:RESult:DPHY:FRAMe<m>:PACKet<n>:IDX? 2271
SEARch:RESult:DPHY:FRAMe<m>:PACKet<n>:VALue? 2271
SEARch:RESult:DPHY:FRAMe<m>:STARt? 2272
SEARch:RESult:DPHY:FRAMe<m>:STATe? 2272
SEARch:RESult:DPHY:FRAMe<m>:STOP? 2272
SEARch:RESult:DPHY:FRAMe<m>:TYPE? 2273
SEARch:RESult:DPHY:FRAMe<m>:VCHannel? 2273

SEARch:RESult:DPHY:FCOunt? <SearchName>

Returns the number of frames within the search result for the selected serial bus.

Query parameters:
<SearchName>

Return values:

<FrameCount>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:DPHY:FRAMe<m>:CS? <SearchName>

Returns the checksum for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<CS>	Range:	0 to 65535
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:DPHY:FRAMe<m>:DATA? <SearchName>

Returns the data or word count value for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<WordCountOrData>	Range:	0 to 65535
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:DPHY:FRAMe<m>:DTName? <SearchName>

Returns the data type name for the specified frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<DataTypeName>

Usage: Query only

SEARch:RESult:DPHY:FRAMe<m>:DTYPe? <SearchName>

Returns the data type for the specified frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<DataType>	Range:	0 to 255
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:DPHY:FRAMe<m>:ECC? <SearchName>

Returns the error correction code (ECC) of the specified frame within the search result.

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<ECC>	Range:	0 to 255
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:DPHY:FRAMe<m>:PACKet<n>:IDX? <SearchName>

Returns the index of the data payload for the selected frame within the search result.

Suffix:

<m> *

<n> *

Query parameters:

<SearchName>

Return values:

<PacketIndex>	Range:	0 to 4294967295
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:DPHY:FRAMe<m>:PACKet<n>:VALue? <SearchName>

Returns the byte value of the data payload for the selected frame within the search result.

Suffix:

<m> *

<n> *

Query parameters:

<SearchName>

Return values:

<PacketValue> Range: 0 to 9223372036854775808
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:DPHY:FRAMe<m>:STARt? <SearchName>

Returns the start time of the specified frame within the search result.

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:DPHY:FRAMe<m>:STATe? <SearchName>

Returns the overall state of the selected frame within the search result.

Suffix:

<m> *

Query parameters:

<SearchName>

Return values:

<State> OK | CS | ECC | ECCWarning | LENGth | FORMat | UNKNOWN_FRAME | INComplete

OK
The frame is valid.

LENGth
Length error.

UNKNOWN_FRAME
Unknown frame type

*RST: OK

Usage: Query only

SEARch:RESult:DPHY:FRAMe<m>:STOP? <SearchName>

Returns the end time for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:DPHY:FRAMe<m>:TYPE? <SearchName>

Returns the type of frame for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameType> HS_SP | HS_LP | HS_INCOMPLETE | LP_HSREQ | LP_TA | LP_ESC

HS_SP
High speed short packet frame

HS_LP
High speed long packet frame

HS_INCOMPLETE
Incomplete high speed frame

LP_HSREQ
High speed request frame

LP_TA
Low power lane turnaround frame

LP_ESC
Low power escape mode frame

*RST: HS_SP

Usage: Query only

SEARch:RESult:DPHY:FRAMe<m>:VCHannel? <SearchName>

Returns the number of virtual channels for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<VirtualChannel> Range: 0 to 3
Increment: 1
*RST: 0

Usage: Query only

24.17.25 M-PHY (option R&S RTO6-K580)

- Configuration....2274
- Trigger M-PHY....2277
- Decode results M-PHY....2292
- Search settings M-PHY....2299
- Search results M-PHY....2315
- Trigger USB SSIC (Option R&S RTO6-K570)....2320
- Search settings USB SSIC (Option R&S RTO6-K570)....2325

24.17.25.1 Configuration

BUS<m>:MPHY:VARiant....2274
BUS<m>:MPHY:DLANes....2274
BUS<m>:MPHY:DZERo:SOURce....2275
BUS<m>:MPHY:DONE:SOURce....2275
BUS<m>:MPHY:DTWO:SOURce....2275
BUS<m>:MPHY:DTHRee:SOURce....2275
BUS<m>:MPHY:DZERo:THReshold....2275
BUS<m>:MPHY:DONE:THReshold....2275
BUS<m>:MPHY:DTWO:THReshold....2275
BUS<m>:MPHY:DTHRee:THReshold....2275
BUS<m>:MPHY:THCoupling....2276
BUS<m>:MPHY:THPReset....2276
BUS<m>:MPHY:SCRMode....2276
BUS<m>:MPHY:SMLanes....2276

BUS<m>:MPHY:VARiant <Protocol>

Selects the protocol running on the interface.

Suffix:

<m> 1..4
Selects the serial bus.

Parameters:

<Protocol> MPHY | UNIPRO | SSIC
*RST: MPHY

BUS<m>:MPHY:DLANes <DataLaneCount>

Sets the number of data lanes.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<DataLaneCount> Range: 1 to 4
Increment: 1
*RST: 4

BUS<m>:MPHY:DZERo:SOURce <DataSource>

Sets the source for the lane.

Suffix:
<m> 1..4

Parameters:
<DataSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4
*RST: C1W1

BUS<m>:MPHY:DONE:SOURce <DataSource>
BUS<m>:MPHY:DTWO:SOURce <DataSource>
BUS<m>:MPHY:DTHRee:SOURce <DataSource>

Selects the source of the corresponding data lane.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<DataSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | NONE
*RST: DONe: C2W1, DTWo: C3W1, DTHree: C4W1

BUS<m>:MPHY:DZERo:THReshold <ThresholdD0>
BUS<m>:MPHY:DONE:THReshold <ThresholdD1>
BUS<m>:MPHY:DTWO:THReshold <ThresholdD2>
BUS<m>:MPHY:DTHRee:THReshold <ThresholdD3>

Sets the threshold value for the respective lane.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ThresholdD1> Range: 0 to 1
<ThresholdD2> Increment: 0.01
<ThresholdD3> *RST: 0.1
Default unit: V

BUS<m>:MPHY:THCoupling <Coupling>

Enables the same threshold value to be used for all lanes.

When the threshold coupling is disabled, changes to the threshold value only affect the respective lane.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<Coupling> ON | OFF
*RST: ON

BUS<m>:MPHY:THPReset <ThresholdPreset>

Selects the predefined value to preset the threshold value of the data lanes.

Suffix:
<m> 1..4
Selects the serial bus.

Parameters:
<ThresholdPreset> THRES_60MV | THRES_120MV | THRES_240MV | MANual

THRES_60MV | THRES_120MV | THRES_240MV
Sets the threshold value of the data lane to the selected predefined value.

MANual
Automatically switches to this option if you edit the threshold text box manually.

*RST: MANual

BUS<m>:MPHY:SCRMode <SSICScrambleMd>

Sets if the SSIC data is scrambled or unscrambled.

Suffix:
<m> 1..4

Parameters:
<SSICScrambleMd> SCRM | NSCR

`SCRM`: scrambled
`NSCR`: unscrambled

*RST: SCRM

BUS<m>:MPHY:SMLanes <Active>

Enables the display of the results in multiple lanes.

Suffix:
<m> 1..4

Parameters:
<Active> ON | OFF
*RST: ON

24.17.25.2 Trigger M-PHY

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:MPHY:TYPE....2278
TRIGger<m>:MPHY:LCCType....2279
TRIGger<m>:MPHY:LWONe:CONDition....2279
TRIGger<m>:MPHY:LWONe:MIN....2280
TRIGger<m>:MPHY:LWONe:MAX....2280
TRIGger<m>:MPHY:LWTWo:CONDition....2280
TRIGger<m>:MPHY:LWTWo:MIN....2281
TRIGger<m>:MPHY:LWTWo:MAX....2281
TRIGger<m>:MPHY:LWTHree:CONDition....2281
TRIGger<m>:MPHY:LWTHree:MIN....2282
TRIGger<m>:MPHY:LWTHree:MAX....2282
TRIGger<m>:MPHY:LWFour:CONDition....2282
TRIGger<m>:MPHY:LWFour:MIN....2283
TRIGger<m>:MPHY:LWFour:MAX....2283
TRIGger<m>:MPHY:TC....2283
TRIGger<m>:MPHY:DATA:DCON....2284
TRIGger<m>:MPHY:DATA:DMIN....2284
TRIGger<m>:MPHY:DATA:DMAX....2284
TRIGger<m>:MPHY:DATA:ICONdition....2285
TRIGger<m>:MPHY:DATA:IMIN....2285
TRIGger<m>:MPHY:DATA:IMAX....2285
TRIGger<m>:MPHY:FSNumber:CONDition....2285
TRIGger<m>:MPHY:FSNumber:MIN....2286
TRIGger<m>:MPHY:FSNumber:MAX....2286
TRIGger<m>:MPHY:CRC:CONDition....2286
TRIGger<m>:MPHY:CRC:MIN....2287
TRIGger<m>:MPHY:CRC:MAX....2287
TRIGger<m>:MPHY:CREQ....2287
TRIGger<m>:MPHY:RREQ....2288
TRIGger<m>:MPHY:CREDit:CONDition....2288
TRIGger<m>:MPHY:CREDit:MIN....2289
TRIGger<m>:MPHY:CREDit:MAX....2289

TRIGger<m>:MPHY:PBEGin:CONDition 2289
TRIGger<m>:MPHY:PBEGin:MIN 2289
TRIGger<m>:MPHY:PBEGin:MAX 2290
TRIGger<m>:MPHY:PFID:CONDition 2290
TRIGger<m>:MPHY:PFID:MIN 2290
TRIGger<m>:MPHY:PFID:MAX 2291
TRIGger<m>:MPHY:PRPLength 2291
TRIGger<m>:MPHY:ERRor:LCMD 2291
TRIGger<m>:MPHY:ERRor:LENGth 2291
TRIGger<m>:MPHY:ERRor:REServed 2292
TRIGger<m>:MPHY:ERRor:SYMBol 2292
TRIGger<m>:MPHY:ERRor:UNKNown 2292

TRIGger<m>:MPHY:TYPE <Type>

Selects the type of frame to be triggered on.

Suffix:

<m> 1..3

Parameters:

<Type> START | BURST | ADAPT | LCC | DLPDUSOF | DLPDUCOF | DLPDUEOF | DLPDUNAC | DLPDUAFC | PACP | TRIGUPR0 | TRIGUPR1 | TRIGUPR2 | ERRor

START
M-PHY or UniPro start of frame.

BURST
M-PHY burst frame.

ADAPT
M-PHY Adapt frame. This control frame is used for clock/bit synchronization.

LCC
M-PHY LCC frame. This control frame is used for line configuration depending on the `LCC Type`.

DLPDUSOF | DLPDUCOF | DLPDUEOF | DLPDUNAC | DLPDUAFC
UniPro DL PDU frame.

PACP
UniPro PACP frame. This control frame is used mainly for power mode change and L1.5 link management.

TRIGUPR0 | TRIGUPR1 | TRIGUPR2
UniPro Trigger Upper frame. This control frame is used for link startup sequence.

ERRor
M-PHY or UniPro error frame.

*RST: START

TRIGger<m>:MPHY:LCCType <LCCType>

Selects the type of LCC frame to be triggered on.

Suffix:
<m> 1..3

Parameters:
<LCCType> ANY | SLEEP | STALL | READCAP | READMFG | READVEND | WRITEATTR | PWMG0 | PWMG1 | PWMG2 | PWMG3 | PWMG4 | PWMG5 | PWMG6 | PWMG7 | HSG1A | HSG2A | HSG3A | HSG4A | HSG1B | HSG2B | HSG3B | HSG4B | RESERVED

ANY
All the available LCC commands.

SLEEP
Switches the power-saving state to ultra-low power. In this state, up to 90% power saving is achieved.

STALL
Switches the power-saving state to ultra-low power. In this state, up to 75% power saving is achieved.

READCAP
Recovers data about the OMC's capabilities.

READMFG
Retrieves manufacturing ID and vendor-specific information.

READVEND
Retrieves the additional four delimited bytes containing vendor-specific information.

WRITEATTR
Sets the configuration parameters required for lane operation.

PWMG0 | PWMG1 | PWMG2 | PWMG3 | PWMG4 | PWMG5 | PWMG6 | PWMG7
Switches the transmission mode to the selected low-power gear.

HSG1A | HSG2A | HSG3A | HSG4A | HSG1B | HSG2B | HSG3B | HSG4B
Switches the transmission mode to the selected high-speed gear.

RESERVED
Reserved bit command. These bits are reserved for future use.

*RST: ANY

TRIGger<m>:MPHY:LWONe:CONDition <CondOperator>

Sets the condition for WORD 1. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<CondOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `TRIGger<m>:MPHY:LWONe:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MPHY:LWONe:MIN` and `TRIGger<m>:MPHY:LWONe:MAX`.

*RST: EQUal

TRIGger<m>:MPHY:LWONe:MIN <LCCWORD1Min>

Specifies the WORD 1 pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<LCCWORD1Min> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:LWONe:MAX <LCCWORD1Max>

Sets the end value of the WORD 1 range if `TRIGger<m>:MPHY:LWONe:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<LCCWORD1Max> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:LWTWo:CONDition <CondOperator>

Sets the condition for WORD 2. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<CondOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `TRIGger<m>:MPHY:LWTWo:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MPHY:LWTWo:MIN` and `TRIGger<m>:MPHY:LWTWo:MAX`.

*RST: EQUal

TRIGger<m>:MPHY:LWTWo:MIN <LCCWORD2Min>

Specifies the WORD 2 pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<LCCWORD2Min> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:LWTWo:MAX <LCCWORD2Max>

Sets the end value of the WORD 2 range if `TRIGger<m>:MPHY:LWTWo:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<LCCWORD2Max> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:LWTHree:CONDition <CondOperator>

Sets the condition for WORD 3. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<CondOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `TRIGger<m>:MPHY:LWTHree:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MPHY:LWTHree:MIN` and `TRIGger<m>:MPHY:LWTHree:MAX`.

*RST: EQUal

TRIGger<m>:MPHY:LWTHree:MIN <LCCWORD3Min>

Specifies the WORD 3 pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<LCCWORD3Min> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:LWTHree:MAX <LCCWORD3Max>

Sets the end value of the WORD 3 range if `TRIGger<m>:MPHY:LWTHree:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<LCCWORD3Max> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:LWFour:CONDition <CondOperator>

Sets the condition for WORD 4. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<CondOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `TRIGger<m>:MPHY:LWFour:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MPHY:LWFour:MIN` and `TRIGger<m>:MPHY:LWFour:MAX`.

*RST: EQUal

TRIGger<m>:MPHY:LWFour:MIN <LCCWORD4Min>

Specifies the WORD 4 pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<LCCWORD4Min> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:LWFour:MAX <LCCWORD4Max>

Sets the end value of the WORD 4 range if `TRIGger<m>:MPHY:LWFour:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<LCCWORD4Max> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:TC <TC>

Specifies the TC bit pattern.

Suffix:
<m> 1..3

Parameters:

<TC> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:DATA:DCON <DataOperator>

Sets the condition for the data. You can define an exact value or a value range.

Suffix:

<m> 1..3

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:MPHY:DATA:DMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MPHY:DATA:DMIN` and `TRIGger<m>:MPHY:DATA:DMAX`.

*RST: EQUal

TRIGger<m>:MPHY:DATA:DMIN <DataMin>

Specifies the data bit pattern, or sets the start value of a pattern range.

Suffix:

<m> 1..3

Parameters:

<DataMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:DATA:DMAX <DataMax>

Sets the end value of the data range if `TRIGger<m>:MPHY:DATA:DCON` is set to `INRange` or `OORange`.

Suffix:

<m> 1..3

Parameters:

<DataMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:DATA:ICONdition <DataIdxOperator>

Sets the condition for the index. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | LTHan | LETHan | GTHan | GETHan
Equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `TRIGger<m>:MPHY:DATA:IMIN` command.

INRange = RANGe
In range: Set the minimum and maximum value using `TRIGger<m>:MPHY:DATA:IMIN` and `TRIGger<m>:MPHY:DATA:IMAX`.

*RST: INRange

TRIGger<m>:MPHY:DATA:IMIN <DataIndexMin>

Specifies the index minimum, or sets the start value of a range.

Suffix:
<m> 1..3

Parameters:
<DataIndexMin> Range: 1 to 0
Increment: 1
*RST: 1

TRIGger<m>:MPHY:DATA:IMAX <DataIndexMax>

Sets the end value of the index range if `TRIGger<m>:MPHY:DATA:ICONdition` is set to `INRange` or `RANge`.

Suffix:
<m> 1..3

Parameters:
<DataIndexMax> Range: 1 to 0
Increment: 1
*RST: 0

TRIGger<m>:MPHY:FSNumber:CONDition <FSNOperator>

Sets the condition for FSN. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<FSNOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:MPHY:FSNumber:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MPHY:FSNumber:MIN` and `TRIGger<m>:MPHY:FSNumber:MAX`.

*RST: EQUal

TRIGger<m>:MPHY:FSNumber:MIN <FSNMin>

Specifies the FSN bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<FSNMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:FSNumber:MAX <FSNMax>

Sets the end value of the FSN range if `TRIGger<m>:MPHY:FSNumber:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<FSNMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:CRC:CONDition <CRCOperator>

Sets the condition for CRC. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<CRCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:MPHY:CRC:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MPHY:CRC:MIN` and `TRIGger<m>:MPHY:CRC:MAX`.

*RST: EQUal

TRIGger<m>:MPHY:CRC:MIN <CRCMin>

Specifies the CRC bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<CRCMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:CRC:MAX <CRCMax>

Sets the end value of the CRC range if `TRIGger<m>:MPHY:CRC:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<CRCMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:CREQ <REQ>

Specifies the CREQ bit pattern.

Suffix:
<m> 1..3

Parameters:
<REQ> ONE | ZERO | DC

ONE
Sets the trigger condition to binary 1.

ZERO
Sets the trigger condition to binary 0.

DC
Sets the trigger condition to any CREQ value (don't care).

*RST: DC

TRIGger<m>:MPHY:RREQ <REQ>

Specifies the RREQ bit pattern.

Suffix:
<m> 1..3

Parameters:
<REQ> ONE | ZERO | DC

ONE
Sets the trigger condition to binary 1.

ZERO
Sets the trigger condition to binary 0.

DC
Sets the trigger condition to any RREQ value (don't care).

*RST: DC

TRIGger<m>:MPHY:CREDit:CONDition <CreditOperator>

Sets the condition for Credit. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<CreditOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:MPHY:CREDit:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MPHY:CREDit:MIN` and `TRIGger<m>:MPHY:CREDit:MAX`.

*RST: EQUal

TRIGger<m>:MPHY:CREDit:MIN <CreditMin>

Specifies the Credit bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<CreditMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:CREDit:MAX <CreditMax>

Sets the end value of the Credit range if `TRIGger<m>:MPHY:CREDit:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<CreditMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:PBEGin:CONDition <CondPACPBegin>

Sets the condition for PACP Begin. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<CondPACPBegin> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:MPHY:PBEGin:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MPHY:PBEGin:MIN` and `TRIGger<m>:MPHY:PBEGin:MAX`.

*RST: EQUal

TRIGger<m>:MPHY:PBEGin:MIN <PACPBEGINMin>

Specifies the PACP Begin bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:
<PACPBEGINMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:PBEGin:MAX <PACPBEGINMax>

Sets the end value of the PACP Begin range if `TRIGger<m>:MPHY:PBEGin:CONDition` is set to `INRange` or `OORange`.

Suffix:
<m> 1..3

Parameters:
<PACPBEGINMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:PFID:CONDition <CondPACPFunc>

Sets the condition for PACP Function ID. You can define an exact value or a value range.

Suffix:
<m> 1..3

Parameters:
<CondPACPFunc> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `TRIGger<m>:MPHY:PFID:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `TRIGger<m>:MPHY:PFID:MIN` and `TRIGger<m>:MPHY:PFID:MAX`.

*RST: EQUal

TRIGger<m>:MPHY:PFID:MIN <PACPFunIDMin>

Specifies the PACP Function ID bit pattern, or sets the start value of a pattern range.

Suffix:
<m> 1..3

Parameters:

<PACPFunIDMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:PFID:MAX <PACPFunIDMax>

Sets the end value of the PACP Function ID range if `TRIGger<m>:MPHY:PFID:CONDition` is set to `INRange` or `OORange`.

Suffix:

<m> 1..3

Parameters:

<PACPFunIDMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:MPHY:PRPLength <Width>

Sets the width for the width trigger based upon the data rate.

Suffix:

<m> 1..3

Parameters:

<Width>
Range: 100E-12 to 10000
Increment: 100E-9
*RST: 5E-9
Default unit: s

TRIGger<m>:MPHY:ERRor:LCMD <LCCError>

Enables/disables trigger on LCC error.

Suffix:

<m> 1..3

Parameters:

<LCCError> ON | OFF
*RST: ON

TRIGger<m>:MPHY:ERRor:LENGth <ErrorLength>

Enables/disables trigger on length error.

Suffix:

<m> 1..3

Parameters:
<ErrorLength> ON | OFF
*RST: ON

TRIGger<m>:MPHY:ERRor:REServed <ErrorReserved>

Enables/disables trigger on reserved bit error.

Suffix:
<m> 1..3

Parameters:
<ErrorReserved> ON | OFF
*RST: ON

TRIGger<m>:MPHY:ERRor:SYMBol <ErrorSymbol>

Enables/disables trigger on 8b10b symbol error.

Suffix:
<m> 1..3

Parameters:
<ErrorSymbol> ON | OFF
*RST: ON

TRIGger<m>:MPHY:ERRor:UNKNown <ErrorUnknown>

Enables/disables trigger on unidentified UniPro frame error.

Suffix:
<m> 1..3

Parameters:
<ErrorUnknown> ON | OFF
*RST: ON

24.17.25.3 Decode results M-PHY

To show the results on the screen, use the following commands:

- `BUS<m>:RESult` on page 1671

BUS<m>:MPHY:RESult:FRAMe<n>:FTYPe?....2293
BUS<m>:MPHY:RESult:FRAMe<n>:STATe?....2294
BUS<m>:MPHY:RESult:FCOunt?....2294
BUS<m>:MPHY:RESult:FRAMe<n>:CCOunt?....2295
BUS<m>:MPHY:RESult:FRAMe<n>:CELL<o>:DATA?....2295
BUS<m>:MPHY:RESult:FRAMe<n>:CELL<o>:STARt?....2295
BUS<m>:MPHY:RESult:FRAMe<n>:CELL<o>:STOP?....2296
BUS<m>:MPHY:RESult:FRAMe<n>:CELL<o>:STATe?....2296

BUS<m>:MPHY:RESult:FRAMe<n>:CELL<o>:TYPE?.....2297
BUS<m>:MPHY:RESult:FRAMe<n>:DATA?.....2297
BUS<m>:MPHY:RESult:FRAMe<n>:INFO?.....2297
BUS<m>:MPHY:RESult:FRAMe<n>:SSFT?.....2298
BUS<m>:MPHY:RESult:FRAMe<n>:STARt?.....2298
BUS<m>:MPHY:RESult:FRAMe<n>:STOP?.....2298

BUS<m>:MPHY:RESult:FRAMe<n>:FTYPe?

Returns the type of frame for the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<FrameType> BURST | ADAPT | LCC | DLPDUSOF | DLPDUCOF | DLPDUEOF | DLPDUNAC | DLPDUAFC | PACP | TRIGUPR0 | TRIGUPR1 | TRIGUPR2 | SKIP | FILLER | SPACER | UNKNOWN

BURST
M-PHY burst frame.

ADAPT
M-PHY Adapt frame. This control frame is used for clock/bit synchronization.

LCC
M-PHY LCC frame. This control frame is used for line configuration depending on the `LCC Type`.

DLPDUSOF | DLPDUCOF | DLPDUEOF | DLPDUNAC | DLPDUAFC
UniPro DL PDU frame.

PACP
UniPro frame. This control frame is used mainly for power mode change and L1.5 link management.

TRIGUPR0 | TRIGUPR1 | TRIGUPR2
UniPro frame. This control frame is used for link startup sequence.

SKIP
UniPro frame with Skip symbols for reducing protocol payload bandwidth.

FILLER
M-PHY or UniPro frame with filler words to maintain transmission activity.

SPACER
Scrambled UniPro frame with filler words.

UNKNOWN
No meaningful frame can be determined.

*RST: UNKNOWN

Usage: Query only

BUS<m>:MPHY:RESult:FRAMe<n>:STATe?

Returns the state of the frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<State> OK | RSVDERROR | CMDERROR | SYMBOLERROR | UNKNOWNERROR | INVALID | INCOMPLETE | UNCorrelated | CRCERROR | VOOR | LENGTHERROR

OK
Valid frame.

RSVDERROR
Erroneous frame due to reserved bit error.

CMDERROR
Erroneous frame due to LCC error.

SYMBOLERROR
Erroneous frame due to 8b10b symbol error.

UNKNOWNERROR
Erroneous frame due to unidentified UniPro frame error.

INVALID
Invalid frame.

INCOMPLETE
The frame is not complete.

LENGTHERROR
Erroneous frame due to length error.

*RST: OK

Usage: Query only

BUS<m>:MPHY:RESult:FCOunt?

Returns the number of decoded frames.

Suffix:

<m> 1..4
Selects the serial bus.

Return values:

<Count> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:MPHY:RESult:FRAMe<n>:CCOunt?

Returns the number of decoded cells for the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<NumWords> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:MPHY:RESult:FRAMe<n>:CELL<o>:DATA?

Returns the data value for the selected cell.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the cell.

Return values:

<WordValue> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:MPHY:RESult:FRAMe<n>:CELL<o>:STARt?

Returns the start time for the selected cell.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the cell.

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:MPHY:RESult:FRAMe<n>:CELL<o>:STOP?

Returns the stop time for the selected cell.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the cell.

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:MPHY:RESult:FRAMe<n>:CELL<o>:STATe?

Returns the state for the selected cell.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the cell.

Return values:

<State> OK | RSVDERROR | CMDERROR | SYMBOLERROR | UNKNOWNERROR | INVALID | INCOMPLETE | UNCorrelated | CRCERROR | VOOR | LENGTHERROR

*RST: OK

Usage: Query only

BUS<m>:MPHY:RESult:FRAMe<n>:CELL<o>:TYPE?

Returns the data type for the selected cell.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

<o> *
Selects the cell.

Return values:
<WordType>

Usage: Query only

BUS<m>:MPHY:RESult:FRAMe<n>:DATA?

Returns the data for the selected frame, corresponds to the Data column in the decode results table.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<Data>

Usage: Query only

BUS<m>:MPHY:RESult:FRAMe<n>:INFO?

Returns the selected frame label. This label is on top of the frame as seen in the honeycomb display, also corresponds to the Info column in the decode results table.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:
<Info>

Usage: Query only

BUS<m>:MPHY:RESult:FRAMe<n>:SSFT?

Returns the type of frame for the selected SSIC frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<SSICFrameType> TSEQ | TS1 | TS1A | TS1B | TS2 | SLF | U2IT | VDT | PCAP | PCNF | PCNR | LDM | ACK | NRDY | ERDY | STATus | STALI | FWAKe | LTM | BIAM | HRR | SSPD | PING | PNGR | DPH | DPHN | DPP | DPPA | ITP | LC | SKIP | BRST | BDAT | BERC | BCNT | IDLE | CP0 | CP1 | CP2 | CP3 | CP5or6 | UHP | ULMP | UTP | UDVN | UDEF | ERRor | CTLB | DATB | SYNC | SDS

*RST: UDEF

Usage: Query only

BUS<m>:MPHY:RESult:FRAMe<n>:STARt?

Returns the start time of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:MPHY:RESult:FRAMe<n>:STOP?

Returns the end time of the selected frame.

Suffix:

<m> 1..4
Selects the serial bus.

<n> *
Selects the frame.

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

24.17.25.4 Search settings M-PHY

SEARch:TRIGger:MPHY:TYPE 2300
SEARch:TRIGger:MPHY:LCCType 2301
SEARch:TRIGger:MPHY:LWONe:CONDition 2302
SEARch:TRIGger:MPHY:LWONe:MIN 2302
SEARch:TRIGger:MPHY:LWONe:MAX 2302
SEARch:TRIGger:MPHY:LWTWo:CONDition 2303
SEARch:TRIGger:MPHY:LWTWo:MIN 2303
SEARch:TRIGger:MPHY:LWTWo:MAX 2303
SEARch:TRIGger:MPHY:LWTHree:CONDition 2304
SEARch:TRIGger:MPHY:LWTHree:MIN 2304
SEARch:TRIGger:MPHY:LWTHree:MAX 2304
SEARch:TRIGger:MPHY:LWFour:CONDition 2305
SEARch:TRIGger:MPHY:LWFour:MIN 2305
SEARch:TRIGger:MPHY:LWFour:MAX 2305
SEARch:TRIGger:MPHY:TC 2306
SEARch:TRIGger:MPHY:DATA:DCONdition 2306
SEARch:TRIGger:MPHY:DATA:DMIN 2306
SEARch:TRIGger:MPHY:DATA:DMAX 2307
SEARch:TRIGger:MPHY:DATA:ICONdition 2307
SEARch:TRIGger:MPHY:DATA:IMIN 2307
SEARch:TRIGger:MPHY:DATA:IMAX 2308
SEARch:TRIGger:MPHY:FSNumber:CONDition 2308
SEARch:TRIGger:MPHY:FSNumber:MIN 2308
SEARch:TRIGger:MPHY:FSNumber:MAX 2309
SEARch:TRIGger:MPHY:CRC:CONDition 2309
SEARch:TRIGger:MPHY:CRC:MIN 2309
SEARch:TRIGger:MPHY:CRC:MAX 2310
SEARch:TRIGger:MPHY:CREQ 2310
SEARch:TRIGger:MPHY:RREQ 2310
SEARch:TRIGger:MPHY:CREDit:CONDition 2311
SEARch:TRIGger:MPHY:CREDit:MIN 2311
SEARch:TRIGger:MPHY:CREDit:MAX 2311
SEARch:TRIGger:MPHY:PBEGin:CONDition 2312
SEARch:TRIGger:MPHY:PBEGin:MIN 2312
SEARch:TRIGger:MPHY:PBEGin:MAX 2312
SEARch:TRIGger:MPHY:PFID:CONDition 2313
SEARch:TRIGger:MPHY:PFID:MIN 2313
SEARch:TRIGger:MPHY:PFID:MAX 2313
SEARch:TRIGger:MPHY:ERRor:REServed 2314
SEARch:TRIGger:MPHY:ERRor:LCMD 2314

SEARch:TRIGger:MPHY:ERRor:SYMBol 2314
SEARch:TRIGger:MPHY:ERRor:UNKNown 2314
SEARch:TRIGger:MPHY:ERRor:LENGth 2315

SEARch:TRIGger:MPHY:TYPE <SearchName>,<Type>
SEARch:TRIGger:MPHY:TYPE? <SearchName>

Selects the type of frame to be searched for.

Parameters:

<Type>

START | BURST | ADAPT | LCC | DLPDUSOF | DLPDUCOF | DLPDUEOF | DLPDUNAC | DLPDUAFC | PACP | TRIGUPR0 | TRIGUPR1 | TRIGUPR2 | ERRor

START
M-PHY or UniPro start of frame.

BURST
M-PHY burst frame.

ADAPT
M-PHY Adapt frame. This control frame is used for clock/bit synchronization.

LCC
M-PHY LCC frame. This control frame is used for line configuration depending on the `LCC Type`.

DLPDUSOF | DLPDUCOF | DLPDUEOF | DLPDUNAC | DLPDUAFC
UniPro DL PDU frame.

PACP
UniPro PACP frame. This control frame is used mainly for power mode change and L1.5 link management.

TRIGUPR0 | TRIGUPR1 | TRIGUPR2
UniPro Trigger Upper frame. This control frame is used for link startup sequence.

ERRor
M-PHY or UniPro error frame.

*RST: START

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:LCCType <SearchName>,<LCCType>
SEARch:TRIGger:MPHY:LCCType? <SearchName>

Selects the type of LCC frame to be searched for.

Parameters:

<LCCType> ANY | SLEEP | STALL | READCAP | READMFG | READVEND | WRITEATTR | PWMG0 | PWMG1 | PWMG2 | PWMG3 | PWMG4 | PWMG5 | PWMG6 | PWMG7 | HSG1A | HSG2A | HSG3A | HSG4A | HSG1B | HSG2B | HSG3B | HSG4B | RESERVED

ANY
All the available LCC commands.

SLEEP
Switches the power-saving state to ultra-low power. In this state, up to 90% power saving is achieved.

STALL
Switches the power-saving state to ultra-low power. In this state, up to 75% power saving is achieved.

READCAP
Recovers data about the OMC's capabilities.

READMFG
Retrieves manufacturing ID and vendor-specific information.

READVEND
Retrieves the additional four delimited bytes containing vendor-specific information.

WRITEATTR
Sets the configuration parameters required for lane operation.

PWMG0 | PWMG1 | PWMG2 | PWMG3 | PWMG4 | PWMG5 | PWMG6 | PWMG7
Switches the transmission mode to the selected low-power gear.

HSG1A | HSG2A | HSG3A | HSG4A | HSG1B | HSG2B | HSG3B | HSG4B
Switches the transmission mode to the selected high-speed gear.

RESERVED
Reserved bit command. These bits are reserved for future use.

*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:LWONe:CONDition <SearchName>,<CondOperator>
SEARch:TRIGger:MPHY:LWONe:CONDition? <SearchName>

Sets the condition for WORD 1. You can define an exact value or a value range.

Parameters:

<CondOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `SEARch:TRIGger:MPHY:LWONe:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:MPHY:LWONe:MIN` and `SEARch:TRIGger:MPHY:LWONe:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:LWONe:MIN <SearchName>,<LCCWORD1Min>
SEARch:TRIGger:MPHY:LWONe:MIN? <SearchName>

Specifies the WORD 1 pattern, or sets the the start value of a pattern range.

Parameters:

<LCCWORD1Min> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:LWONe:MAX <SearchName>,<LCCWORD1Max>
SEARch:TRIGger:MPHY:LWONe:MAX? <SearchName>

Sets the end value of the WORD 1 range if `SEARch:TRIGger:MPHY:LWONe:CONDition` is set to `INRange` or `OORange`.

Parameters:

<LCCWORD1Max> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:LWTWo:CONDition <SearchName>,<CondOperator>
SEARch:TRIGger:MPHY:LWTWo:CONDition? <SearchName>

Sets the condition for WORD 2. You can define an exact value or a value range.

Parameters:

<CondOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `SEARch:TRIGger:MPHY:LWTWo:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:MPHY:LWTWo:MIN` and `SEARch:TRIGger:MPHY:LWTWo:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:LWTWo:MIN <SearchName>,<LCCWORD2Min>
SEARch:TRIGger:MPHY:LWTWo:MIN? <SearchName>

Specifies the WORD 2 pattern, or sets the the start value of a pattern range.

Parameters:

<LCCWORD2Min> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:LWTWo:MAX <SearchName>,<LCCWORD2Max>
SEARch:TRIGger:MPHY:LWTWo:MAX? <SearchName>

Sets the end value of the WORD 2 range if `SEARch:TRIGger:MPHY:LWTWo:CONDition` is set to `INRange` or `OORange`.

Parameters:

<LCCWORD2Max> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:LWTHree:CONDition <SearchName>,<CondOperator>
SEARch:TRIGger:MPHY:LWTHree:CONDition? <SearchName>

Sets the condition for WORD 3. You can define an exact value or a value range.

Parameters:

<CondOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `SEARch:TRIGger:MPHY:LWTHree:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:MPHY:LWTHree:MIN` and `SEARch:TRIGger:MPHY:LWTHree:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:LWTHree:MIN <SearchName>,<LCCWORD3Min>
SEARch:TRIGger:MPHY:LWTHree:MIN? <SearchName>

Specifies the WORD 3 pattern, or sets the the start value of a pattern range.

Parameters:

<LCCWORD3Min> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:LWTHree:MAX <SearchName>,<LCCWORD3Max>
SEARch:TRIGger:MPHY:LWTHree:MAX? <SearchName>

Sets the end value of the WORD 3 range if `SEARch:TRIGger:MPHY:LWTHree:CONDition` is set to `INRange` or `OORange`.

Parameters:

<LCCWORD3Max> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:LWFour:CONDition <SearchName>,<CondOperator>
SEARch:TRIGger:MPHY:LWFour:CONDition? <SearchName>

Sets the condition for WORD 4. You can define an exact value or a value range.

Parameters:

<CondOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `SEARch:TRIGger:MPHY:LWFour:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:MPHY:LWFour:MIN` and `SEARch:TRIGger:MPHY:LWFour:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:LWFour:MIN <SearchName>,<LCCWORD4Min>
SEARch:TRIGger:MPHY:LWFour:MIN? <SearchName>

Specifies the WORD 4 pattern, or sets the the start value of a pattern range.

Parameters:

<LCCWORD4Min> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:LWFour:MAX <SearchName>,<LCCWORD4Max>
SEARch:TRIGger:MPHY:LWFour:MAX? <SearchName>

Sets the end value of the WORD 4 range if `SEARch:TRIGger:MPHY:LWFour:CONDition` is set to `INRange` or `OORange`.

Parameters:

<LCCWORD4Max> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:TC <SearchName>,<TC>
SEARch:TRIGger:MPHY:TC? <SearchName>

Specifies the TC bit pattern.

Parameters:

<TC> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:DATA:DCONdition <SearchName>,<DataOperator>
SEARch:TRIGger:MPHY:DATA:DCONdition? <SearchName>

Sets the condition for the data. You can define an exact value or a value range.

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:MPHY:DATA:DMIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:MPHY:DATA:DMIN` and `SEARch:TRIGger:MPHY:DATA:DMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:DATA:DMIN <SearchName>,<DataMin>
SEARch:TRIGger:MPHY:DATA:DMIN? <SearchName>

Specifies the data bit pattern, or sets the the start value of a pattern range.

Parameters:

<DataMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:DATA:DMAX <SearchName>,<DataMax>
SEARch:TRIGger:MPHY:DATA:DMAX? <SearchName>

Sets the end value of the data range if `SEARch:TRIGger:MPHY:DATA:DCONdition` is set to `INRange` or `OORange`.

Parameters:

<DataMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:DATA:ICONdition <SearchName>,<DataIdxOperator>
SEARch:TRIGger:MPHY:DATA:ICONdition? <SearchName>

Sets the condition for the index. You can define an exact value or a value range.

Parameters:

<DataIdxOperator> EQUal | LTHan | LETHan | GTHan | GETHan | INRange | RANGe

EQUal | LTHan | LETHan | GTHan | GETHan
Equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one pattern to be set with the corresponding `SEARch:TRIGger:MPHY:DATA:IMIN` command.

INRange = RANGe
In range: Set the minimum and maximum value using `SEARch:TRIGger:MPHY:DATA:IMIN` and `SEARch:TRIGger:MPHY:DATA:IMAX`.

*RST: INRange

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:DATA:IMIN <SearchName>,<DataIndexMin>
SEARch:TRIGger:MPHY:DATA:IMIN? <SearchName>

Specifies the index minimum, or sets the start value of a range.

Parameters:

<DataIndexMin> Range: 1 to 0
Increment: 1
*RST: 1

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:DATA:IMAX <SearchName>,<DataIndexMax>
SEARch:TRIGger:MPHY:DATA:IMAX? <SearchName>

Sets the end value of the index range if `SEARch:TRIGger:MPHY:DATA:ICONdition` is set to `INRange` or `RANge`.

Parameters:

<DataIndexMax> Range: 1 to 0
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:FSNumber:CONDition <SearchName>,<FSNOperator>
SEARch:TRIGger:MPHY:FSNumber:CONDition? <SearchName>

Sets the condition for FSN. You can define an exact value or a value range.

Parameters:

<FSNOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:MPHY:FSNumber:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:MPHY:FSNumber:MIN` and `SEARch:TRIGger:MPHY:FSNumber:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:FSNumber:MIN <SearchName>,<FSNMin>
SEARch:TRIGger:MPHY:FSNumber:MIN? <SearchName>

Specifies the FSN bit pattern, or sets the start value of a pattern range.

Parameters:

<FSNMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:FSNumber:MAX <SearchName>,<FSNMax>
SEARch:TRIGger:MPHY:FSNumber:MAX? <SearchName>

Sets the end value of the FSN range if `SEARch:TRIGger:MPHY:FSNumber:CONDition` is set to `INRange` or `OORange`.

Parameters:

<FSNMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:CRC:CONDition <SearchName>,<CRCOperator>
SEARch:TRIGger:MPHY:CRC:CONDition? <SearchName>

Sets the condition for CRC. You can define an exact value or a value range.

Parameters:

<CRCOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:MPHY:CRC:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:MPHY:CRC:MIN` and `SEARch:TRIGger:MPHY:CRC:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:CRC:MIN <SearchName>,<CRCMin>
SEARch:TRIGger:MPHY:CRC:MIN? <SearchName>

Specifies the CRC bit pattern, or sets the start value of a pattern range.

Parameters:

<CRCMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:CRC:MAX <SearchName>,<CRCMax>
SEARch:TRIGger:MPHY:CRC:MAX? <SearchName>

Sets the end value of the CRC range if `SEARch:TRIGger:MPHY:CRC:CONDition` is set to `INRange` or `OORange`.

Parameters:

<CRCMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:CREQ <SearchName>,<REQ>
SEARch:TRIGger:MPHY:CREQ? <SearchName>

Specifies the CREQ bit pattern.

Parameters:

<REQ> ONE | ZERO | DC

ONE
Sets the trigger condition to binary 1.

ZERO
Sets the trigger condition to binary 0.

DC
Sets the trigger condition to any CREQ value (don't care).

*RST: DC

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:RREQ <SearchName>,<REQ>
SEARch:TRIGger:MPHY:RREQ? <SearchName>

Specifies the RREQ bit pattern.

Parameters:

<REQ> ONE | ZERO | DC

ONE
Sets the trigger condition to binary 1.

ZERO
Sets the trigger condition to binary 0.

DC
Sets the trigger condition to any RREQ value (don't care).

*RST: DC

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:CREDit:CONDition <SearchName>,<CreditOperator>
SEARch:TRIGger:MPHY:CREDit:CONDition? <SearchName>

Sets the condition for Credit. You can define an exact value or a value range.

Parameters:

<CreditOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:MPHY:CREDit:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:MPHY:CREDit:MIN` and `SEARch:TRIGger:MPHY:CREDit:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:CREDit:MIN <SearchName>,<CreditMin>
SEARch:TRIGger:MPHY:CREDit:MIN? <SearchName>

Specifies the Credit bit pattern, or sets the start value of a pattern range.

Parameters:

<CreditMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:CREDit:MAX <SearchName>,<CreditMax>
SEARch:TRIGger:MPHY:CREDit:MAX? <SearchName>

Sets the end value of the Credit range if `SEARch:TRIGger:MPHY:CREDit:CONDition` is set to `INRange` or `OORange`.

Parameters:

<CreditMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:PBEGin:CONDition <SearchName>,<CondPACPBegin>
SEARch:TRIGger:MPHY:PBEGin:CONDition? <SearchName>

Sets the condition for PACP Begin. You can define an exact value or a value range.

Parameters:

<CondPACPBegin> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:MPHY:PBEGin:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:MPHY:PBEGin:MIN` and `SEARch:TRIGger:MPHY:PBEGin:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:PBEGin:MIN <SearchName>,<PACPBEGINMin>
SEARch:TRIGger:MPHY:PBEGin:MIN? <SearchName>

Specifies the PACP Begin bit pattern, or sets the start value of a pattern range.

Parameters:

<PACPBEGINMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:PBEGin:MAX <SearchName>,<PACPBEGINMax>
SEARch:TRIGger:MPHY:PBEGin:MAX? <SearchName>

Sets the end value of the PACP Begin range if `SEARch:TRIGger:MPHY:PBEGin:CONDition` is set to `INRange` or `OORange`.

Parameters:

<PACPBEGINMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:PFID:CONDition <SearchName>,<CondPACPFunc>
SEARch:TRIGger:MPHY:PFID:CONDition? <SearchName>

Sets the condition for PACP Function ID. You can define an exact value or a value range.

Parameters:

<CondPACPFunc> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less or equal than, Greater Than, Greater or equal than. These conditions require one data pattern to be set with the corresponding `SEARch:TRIGger:MPHY:PFID:MIN` command.

INRange | OORange
In range/Out of range: set the minimum and maximum value of the range with `SEARch:TRIGger:MPHY:PFID:MIN` and `SEARch:TRIGger:MPHY:PFID:MAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:PFID:MIN <SearchName>,<PACPFunIDMin>
SEARch:TRIGger:MPHY:PFID:MIN? <SearchName>

Specifies the PACP Function ID bit pattern, or sets the start value of a pattern range.

Parameters:

<PACPFunIDMin> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:PFID:MAX <SearchName>,<PACPFunIDMax>
SEARch:TRIGger:MPHY:PFID:MAX? <SearchName>

Sets the end value of the PACP Function ID range if `SEARch:TRIGger:MPHY:PFID:CONDition` is set to `INRange` or `OORange`.

Parameters:

<PACPFunIDMax> Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:ERRor:REServed <SearchName>,<ErrorReserved>
SEARch:TRIGger:MPHY:ERRor:REServed? <SearchName>

Enables/disables search for reserved bit error.

Parameters:
<ErrorReserved> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:ERRor:LCMD <SearchName>,<LCCError>
SEARch:TRIGger:MPHY:ERRor:LCMD? <SearchName>

Enables/disables search for LCC error.

Parameters:
<LCCError> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:ERRor:SYMBol <SearchName>,<ErrorSymbol>
SEARch:TRIGger:MPHY:ERRor:SYMBol? <SearchName>

Enables/disables search for 8b10b symbol error.

Parameters:
<ErrorSymbol> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:ERRor:UNKNown <SearchName>,<ErrorUnknown>
SEARch:TRIGger:MPHY:ERRor:UNKNown? <SearchName>

Enables/disables search for unidentified UniPro frame error.

Parameters:
<ErrorUnknown> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:ERRor:LENGth <SearchName>,<ErrorLength>
SEARch:TRIGger:MPHY:ERRor:LENGth? <SearchName>

Enables/disables search for length error.

Parameters:
<ErrorLength> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName>

24.17.25.5 Search results M-PHY

In all `SEARch:RESult:MPHY:FRAMe<m>` commands, the suffix <m> selects the frame number and suffix <n> selects the cell number in the list of search results.

SEARch:RESult:MPHY:FCOunt?....2315
SEARch:RESult:MPHY:FRAMe<m>:CCOunt?....2315
SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:DATA?....2316
SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:TYPE?....2316
SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:STARt?....2316
SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:STOP?....2317
SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:STATe?....2317
SEARch:RESult:MPHY:FRAMe<m>:DATA?....2317
SEARch:RESult:MPHY:FRAMe<m>:FTYPe?....2318
SEARch:RESult:MPHY:FRAMe<m>:STATe?....2319
SEARch:RESult:MPHY:FRAMe<m>:STARt?....2319
SEARch:RESult:MPHY:FRAMe<m>:STOP?....2320

SEARch:RESult:MPHY:FCOunt? <SearchName>

Returns the number of frames within the search result.

Query parameters:
<SearchName>

Return values:
<Count> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:MPHY:FRAMe<m>:CCOunt? <SearchName>

Returns the number of cells within the search result for the selected frame.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<NumWords>	Range:	0 to 4294967295
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:DATA? <SearchName>

Returns the data value within the search result for the selected cell.

Suffix:

<m>	*
<n>	*

Query parameters:
<SearchName>

Return values:

<WordValue>	Range:	0 to 4294967295
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:TYPE? <SearchName>

Returns the data type within the search result for the selected cell.

Suffix:

<m>	*
<n>	*

Query parameters:
<SearchName>

Return values:
<WordType>

Usage: Query only

SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:STARt? <SearchName>

Returns the start time of the cell with the selected field number within the frame with the selected frame number within the search result.

Suffix:

<m>	*
<n>	*

Query parameters:
<SearchName>

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:STOP? <SearchName>

Returns the stop time of the cell with the selected field number within the frame with the selected frame number within the search result.

Suffix:

<m> *

<n> *

Query parameters:

<SearchName>

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:STATe? <SearchName>

Returns the state type within the search result for the selected cell.

Suffix:

<m> *

<n> *

Query parameters:

<SearchName>

Return values:

<State> OK | RSVDERROR | CMDERROR | SYMBOLERROR | UNKNOWNERROR | INVALID | INCOMPLETE | UNCorrelated | CRCERROR | VOOR | LENGTHERROR

*RST: OK

Usage: Query only

SEARch:RESult:MPHY:FRAMe<m>:DATA? <SearchName>

Returns the data for the selected frame within the search result, corresponds to the Data column in the search results table.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:MPHY:FRAMe<m>:FTYPe? <SearchName>

Returns the type of frame for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FrameType> BURST | ADAPT | LCC | DLPDUSOF | DLPDUCOF | DLPDUEOF | DLPDUNAC | DLPDUAFC | PACP | TRIGUPR0 | TRIGUPR1 | TRIGUPR2 | SKIP | FILLER | SPACER | UNKNOWN

BURST
M-PHY burst frame.

ADAPT
M-PHY Adapt frame. This control frame is used for clock/bit synchronization.

LCC
M-PHY LCC frame. This control frame is used for line configuration depending on the `LCC Type`.

DLPDUSOF | DLPDUCOF | DLPDUEOF | DLPDUNAC | DLPDUAFC
UniPro DL PDU frame.

PACP
UniPro frame. This control frame is used mainly for power mode change and L1.5 link management.

TRIGUPR0 | TRIGUPR1 | TRIGUPR2
UniPro frame. This control frame is used for link startup sequence.

SKIP
UniPro frame with Skip symbols for reducing protocol payload bandwidth.

FILLER
M-PHY or UniPro frame with filler words to maintain transmission activity.

SPACER
Scrambled UniPro frame with filler words.

UNKNOWN
No meaningful frame can be determined.

*RST: UNKNOWN

Usage: Query only

SEARch:RESult:MPHY:FRAMe<m>:STATe? <SearchName>

Returns the state of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> OK | RSVDERROR | CMDERROR | SYMBOLERROR | UNKNOWNERROR | INVALID | INCOMPLETE | UNCorrelated | CRCERROR | VOOR | LENGTHERROR

OK
Valid frame.

RSVDERROR
Erroneous frame due to reserved bit error.

CMDERROR
Erroneous frame due to LCC error.

LENGTHERROR
Erroneous frame due to length error.

SYMBOLERROR
Erroneous frame due to 8b10b symbol error.

UNKNOWNERROR
Erroneous frame due to unidentified UniPro frame error.

INVALID
Invalid frame.

INCOMPLETE
The frame is not complete.

*RST: OK

Usage: Query only

SEARch:RESult:MPHY:FRAMe<m>:STARt? <SearchName>

Returns the start time of the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Start>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

SEARch:RESult:MPHY:FRAMe<m>:STOP? <SearchName>

Returns the end time for the selected frame within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Stop>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

24.17.25.6 Trigger USB SSIC (Option R&S RTO6-K570)

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

There are two commands for each parameter, that you can use for defining the SSIC trigger settings.

For example, to set the *Frame type =ACK > Field =Route String >Data* value you can use one of the following commands:

- `TRIGger:MPHY:SSIC:FRAMe13:FLD1:DMIN 01100`
  Defines the parameter by using the index <m> for the frame number and <n> for the field number.
- `TRIGger:MPHY:SSIC:DMIN "ACK", "Route String", 01100`
  Defines the parameter by using the frame and field name.

TRIGger<m>:MPHY:SSIC:ERENable....2321
TRIGger<m>:MPHY:SSIC:ERRor<n>:ENABle....2321
TRIGger<m>:MPHY:SSIC:FRENable....2321
TRIGger<m>:MPHY:SSIC:FRAMe<n>:ENABle....2321
TRIGger<m>:MPHY:SSIC:FIENable....2322
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:ENABle....2322
TRIGger<m>:MPHY:SSIC:BIT....2322
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:BIT....2322
TRIGger<m>:MPHY:SSIC:DMAX....2323
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DMAX....2323
TRIGger<m>:MPHY:SSIC:DMIN....2323
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DMIN....2323
TRIGger<m>:MPHY:SSIC:DOPerator....2323
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DOPerator....2323
TRIGger<m>:MPHY:SSIC:IMAX....2324
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IMAX....2324
TRIGger<m>:MPHY:SSIC:IMIN....2324
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IMIN....2324
TRIGger<m>:MPHY:SSIC:IOPerator....2325
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IOPerator....2325

TRIGger<m>:MPHY:SSIC:ERENable <ErrorName>, <Enabler>
TRIGger<m>:MPHY:SSIC:ERRor<n>:ENABle <Enable>

Enables or disables the trigger on an error condition.

Suffix:

<m> 1..3

<n> *

Parameters:

<Enable> ON | OFF

*RST: ON

Example: Trigger on CRC and length errors:

```
TRIGger:MPHY:SSIC:ERRor1:ENABle ON
```

Enables CRC Error.

```
TRIGger:MPHY:SSIC:ERENable "Length Error",ON
```

Enables Length Error.

TRIGger<m>:MPHY:SSIC:FRENable <Frame>, <Enabler>
TRIGger<m>:MPHY:SSIC:FRAMe<n>:ENABle <Enable>

Enables or disables the checking condition for the selected frame.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the field number within the frame.

Parameters:

<Enable> ON | OFF

*RST: OFF

Example: `TRIGger:MPHY:SSIC:FRAMe1:ENABle ON`
Enables the checking condition for the TSEQ frame.
`TRIGger:MPHY:SSIC:FRENable "TS1" ON`
Enables the checking condition for the TS1 frame.

TRIGger<m>:MPHY:SSIC:FIENable <Frame>, <Field>, <Enabler>
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:ENABle <CondEnabler>

Enables or disables the checking condition for the selected field of the selected frame.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the field number within the frame.

<o> *
Specifies the field number within the frame.

Parameters:

<CondEnabler> ON | OFF

*RST: OFF

Example: `TRIGger:MPHY:SSIC:FRAMe1:FLD1:ENABle ON`
Enables the checking condition for the Symbol field of the TSEQ frame.
`TRIGger:MPHY:SSIC:FRENable "TS1", "Loop" ON`
Enables the checking condition for the Loop field of the TS1 frame.

TRIGger<m>:MPHY:SSIC:BIT <Frame>, <Field>, <Bit>
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:BIT <BitState>

Sets the bit state of a field that only consists of one bit.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the frame number.

<o> *
Specifies the field number within the frame.

Parameters:

<BitState> ONE | ZERO | DC

*RST: DC

TRIGger<m>:MPHY:SSIC:DMAX <Frame>, <Field>, <Data>
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DMAX <DataMax>

Sets the end value of a data pattern range if `TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DOPerator` is set to `INRange` or `OORange`.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the frame number.

<o> *
Specifies the field number within the frame.

Parameters:

<DataMax>

TRIGger<m>:MPHY:SSIC:DMIN <Frame>, <Field>, <Data>
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DMIN <DataMin>

Specifies the data pattern, or sets the start value of a data pattern range.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the frame number.

<o> *
Specifies the field number within the frame.

Parameters:

<DataMin>

TRIGger<m>:MPHY:SSIC:DOPerator <Frame>, <Field>, <Operator>
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DOPerator <DataOperator>

Sets the operator for the data pattern in the selected field of the selected frame.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the field number within the frame.

<o> *

Parameters:

<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one endpoint value to be set using `TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DMIN`.

INRange | OORange
In range, out of range. These conditions require a range of endpoint values to be set using `TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DMIN` and `TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DMAX`.

*RST: EQUal

TRIGger<m>:MPHY:SSIC:IMAX <Frame>, <Field>, <Data>
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IMAX <IndexMax>

Sets the end value of an index range if `TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IOPerator` is set to `INRange` or `RANGe`.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the field number within the frame.

<o> *

Parameters:

<IndexMax> Specifies the end value for the index range within the field. The index range, increment and *RST values depend on the field type.

TRIGger<m>:MPHY:SSIC:IMIN <Frame>, <Field>, <Data>
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IMIN <IndexMin>

Specifies the index, or sets the start value of an index range.

Suffix:

<m> 1..3

<n> *
Specifies the field number within the frame.

<o> *

Parameters:

<IndexMin> Specifies the index value or index start value within the field. The index range, increment and *RST values depend on the field type.

Range: 1 to 65534
Increment: 1
*RST: 1

TRIGger<m>:MPHY:SSIC:IOPerator <Frame>, <Field>, <Operator>
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IOPerator <IndexOperator>

Sets the operator for the index in the selected field of the selected frame.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

<n> *
Specifies the field number within the frame.

<o> *

Parameters:

<IndexOperator> EQUal | INRange | RANGe

EQUal
This condition requires one endpoint value to be set using `TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IMIN` and `TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IMAX`.

*RST: INRange

24.17.25.7 Search settings USB SSIC (Option R&S RTO6-K570)

There are two commands for each parameter, that you can use for defining the SSIC search settings.

For example, to set the *Frame type =ACK* > *Field =Route String* >*Data* value you can use one of the following commands:

- `SEARch:TRIGger:MPHY:SSIC:FRAMe13:FLD1:DMIN 01100`
  Defines the parameter by using the index <m> for the frame number and <n> for the field number.
- `SEARch:TRIGger:MPHY:SSIC:DMIN "ACK", "Route String", 01100`
  Defines the parameter by using the frame and field name.

SEARch:TRIGger:MPHY:SSIC:ERENable <SearchName>,<ErrorName>, <Enabler>
SEARch:TRIGger:MPHY:SSIC:ERRor<m>:ENABle <SearchName>,<Enable>
SEARch:TRIGger:MPHY:SSIC:ERRor<m>:ENABle? <SearchName>

Enables or disables the searching for an error condition.

Suffix:

<m> *

Parameters:

<Enable> ON | OFF

*RST: ON

Parameters for setting and query:
<SearchName>

Example: Searches for CRC and length errors:
`SEARch:TRIGger:MPHY:SSIC:ERRor1:ENABle ON`
Enables searching for CRC Error.
`SEARch:TRIGger:MPHY:SSIC:ERENable "Length Error",ON`
Enables searching for Length Error.

SEARch:TRIGger:MPHY:SSIC:FRENable <SearchName>,<Frame>, <Enabler>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:ENABle <SearchName>,<Enable>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:ENABle? <SearchName>

Defines the frame type to be searched for.

Suffix:
<m> *
Specifies the frame number.

Parameters:
<Enable> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName> String with the name of the search.

Example: `SEARch:TRIGger:MPHY:SSIC:FRAMe1:ENABle ON`
Enables searching for the TSEQ frame.
`SEARch:TRIGger:MPHY:SSIC:FRENable "TS1" ON`
Enables searching for the TS1 frame.

SEARch:TRIGger:MPHY:SSIC:FIENable <SearchName>,<Frame>, <Field>, <Enabler>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:ENABle <SearchName>,<CondEnabler>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:ENABle? <SearchName>

Enables or disables the checking condition for searching a specific data pattern in the selected field of the selected frame.

Suffix:
<m> *
Specifies the frame number.

<n> *
Specifies the field number within the frame.

Parameters:
<CondEnabler> ON | OFF
ON
Checking condition enabled

OFF
Checking condition disabled

*RST: OFF

Parameters for setting and query:
<SearchName>

Example: `SEARch:TRIGger:MPHY:SSIC:FRAMe1:FLD1:ENABle ON`
Enables searching for the Symbol field of the TSEQ frame.
`SEARch:TRIGger:MPHY:SSIC:FRENable "TS1", "Loop" ON`
Enables searching for the Loop field of the TS1 frame.

SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:ENABle <SearchName>,<CondEnabler>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:ENABle? <SearchName>

Enables or disables the checking condition for searching a specific data pattern in the selected field of the selected frame.

Suffix:
<m> *
Specifies the frame number.

<n> *
Specifies the field number within the frame.

Parameters:
<CondEnabler> ON | OFF

ON
Checking condition enabled

OFF
Checking condition disabled

*RST: OFF

Parameters for setting and query:
<SearchName>

Example: `SEARch:TRIGger:MPHY:SSIC:FRAMe1:FLD1:ENABle ON`
Enables searching for the Symbol field of the TSEQ frame.
`SEARch:TRIGger:MPHY:SSIC:FRENable "TS1", "Loop" ON`
Enables searching for the Loop field of the TS1 frame.

SEARch:TRIGger:MPHY:SSIC:BIT <SearchName>,<Frame>, <Field>, <Bit>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:BIT <SearchName>,<BitState>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:BIT? <SearchName>

Sets the bit state of a field to be searched that only consists of one bit.

Suffix:

<m> *
Specifies the frame number.

<n> *
Specifies the field number within the frame.

Parameters:

<BitState> ONE | ZERO | DC

ONE
1

ZERO
0

DC
"Don't care" (DC) = X

*RST: DC

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:MPHY:SSIC:DMAX <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DMAX <SearchName>,<DataMax>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DMAX? <SearchName>

Sets the end value of a data pattern range if `SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DOPerator` is set to `INRange` or `OORange`.

Suffix:

<m> *
Specifies the frame number.

<n> *
Specifies the field number within the frame.

Parameters:

<DataMax> Specifies the name of the generic trigger setting frame.

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:MPHY:SSIC:DMIN <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DMIN <SearchName>,<DataMin>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DMIN? <SearchName>

Specifies the data pattern to be searched, or sets the start value of a data pattern range to be searched.

Suffix:

<m> *
Specifies the frame number.

<n> *
Specifies the field number within the frame.

Parameters:
<DataMin> Specifies the name of the generic trigger setting frame.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:SSIC:DOPerator <SearchName>,<Frame>, <Field>, <Operator>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DOPerator <SearchName>,<DataOperator>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DOPerator? <SearchName>

Sets the operator to set a specific data pattern to be searched in the selected field of the selected frame.

Suffix:
<m> *
Specifies the frame number.

<n> *
Specifies the field number within the frame.

Parameters:
<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one endpoint value to be set using `SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DMIN`.

INRange | OORange
In range, out of range. These conditions require a range of endpoint values to be set using `SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DMIN` and `SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:MPHY:SSIC:FRENable <SearchName>,<Frame>, <Enabler>
Setting parameters:
<SearchName>

<Frame>

<Enabler> ON | OFF

Usage: Setting only

SEARch:TRIGger:MPHY:SSIC:IMAX <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IMAX
<SearchName>,<IndexMax>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IMAX? <SearchName>

Sets the end value of an index range if `SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IOPerator` is set to `INRange` or `RANGe`.

Suffix:

<m> *
Specifies the frame number.

<n> *
Specifies the field number within the frame.

Parameters:

<IndexMax> Specifies the end value for the index range within the field.
The index range, increment and *RST values depend on the field type.

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:MPHY:SSIC:IMIN <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IMIN
<SearchName>,<IndexMin>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IMIN? <SearchName>

Specifies the index at which the data is to be searched, or sets the start value of an index range in which the data is to be searched.

Suffix:

<m> *
Specifies the field number within the frame.

<n> *

Parameters:

<IndexMin> Specifies the index value or the start value of an index range within the field.
The index range, increment and *RST values depend on the field type.

Parameters for setting and query:

<SearchName> String with the name of the search.

SEARch:TRIGger:MPHY:SSIC:IOPerator <SearchName>,<Frame>, <Field>, <Operator>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IOPerator
<SearchName>,<IndexOperator>
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IOPerator? <SearchName>

Sets the operator for the index for searching in the selected field of the selected frame.

Suffix:

<m> *
Specifies the field number within the frame.

<n> *

Parameters:

<IndexOperator> EQUal | INRange | RANGe

EQUal
This condition requires one endpoint value to be set using `SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IMIN` and `SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IMAX`.

*RST: INRange

Parameters for setting and query:

<SearchName>

24.17.26 8B/10B (option R&S RTO6-K590)

24.17.26.1 Configuration

In all `BUS<m>:EBTB` commands, the suffix <m> selects the serial bus.

BUS<m>:EBTB:BITRate....2332
BUS<m>:EBTB:EBTRate?....2332
BUS<m>:EBTB:DIFFerential:SOURce....2332
BUS<m>:EBTB:DIFFerential:THRHigh....2332
BUS<m>:EBTB:DIFFerential:THRLow....2333
BUS<m>:EBTB:DISF....2333
BUS<m>:EBTB:DMINus:SOURce....2333
BUS<m>:EBTB:DMINus:THReshold....2333
BUS<m>:EBTB:DPLus:SOURce....2334
BUS<m>:EBTB:DPLus:THReshold....2334
BUS<m>:EBTB:FCSY....2334
BUS<m>:EBTB:SCOunt?....2334
BUS<m>:EBTB:SCSY....2335
BUS<m>:EBTB:SYNC....2335
BUS<m>:EBTB:TYPE....2335
BUS<m>:EBTB:USCS....2335
BUS<m>:EBTB:BITDetermi....2336
BUS<m>:EBTB:FAUToscale....2336

BUS<m>:EBTB:BITRate <Bitrate>

Sets the number of transmitted bits per second.

Suffix:
<m> 1..4

Parameters:
<Bitrate> Range: 100 to 200E+9
Increment: 10
*RST: 1.25E+9
Default unit: bps

BUS<m>:EBTB:EBTRate?

Queries the estimated bitrate value.

Suffix:
<m> 1..4

Return values:
<EstimatedBitrate> Range: 100 to 100E+9
Increment: 10
*RST: 1.25E+9
Default unit: bps

Usage: Query only

BUS<m>:EBTB:DIFFerential:SOURce <SrcDiff>

Selects the source of the provided differential signal, if `BUS<m>:EBTB:TYPE DIFFerential` is set.

Suffix:
<m> 1..4

Parameters:
<SrcDiff> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4
*RST: C1W1

Usage: Asynchronous command

BUS<m>:EBTB:DIFFerential:THRHigh <ThresholdHigh>

Sets the high threshold of the signal, if `BUS<m>:EBTB:TYPE DIFFerential`.

Suffix:
<m> 1..4

Parameters:

<ThresholdHigh>	Range:	-10 to 10
	Increment:	1E-3
	*RST:	0.15
	Default unit:	V

BUS<m>:EBTB:DIFFerential:THRLow <ThresholdLow>

Sets the low threshold of the signal, if `BUS<m>:EBTB:TYPE DIFFerential`.

Suffix:

<m> 1..4

Parameters:

<ThresholdLow>	Range:	-10 to 10
	Increment:	1E-3
	*RST:	-0.15
	Default unit:	V

BUS<m>:EBTB:DISF <DisplayFormat>

Selects the display format for the results in the decode table and for the results of `BUS<m>:EBTB:SYMBol<n>:DATA?`.

Suffix:

<m> 1..4

Parameters:

<DisplayFormat> KD | EB | TB

KD: K/D codes
EB: 8-bit pattern
TB: 10-bit pattern

*RST: KD

BUS<m>:EBTB:DMINus:SOURce <SourceDminus>

Selects the D- source of the provided single ended signal, if `BUS<m>:EBTB:TYPE SINGle` is set.

Suffix:

<m> 1..4

Parameters:

<SourceDminus> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

Usage: Asynchronous command

BUS<m>:EBTB:DMINus:THReshold <ThresholdDminus>

Sets the low threshold (D-) of the signal, if `BUS<m>:EBTB:TYPE SINGle`.

Suffix:
<m> 1..4

Parameters:
<ThresholdDminus> Range: -10 to 10
Increment: 1E-3
*RST: 0
Default unit: V

BUS<m>:EBTB:DPLus:SOURce <SourceDplus>

Selects the D+ source of the provided single-ended signal, if `BUS<m>:EBTB:TYPE SINGle` is set.

Suffix:
<m> 1..4

Parameters:
<SourceDplus> NONE | C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4
*RST: C1W1

Usage: Asynchronous command

BUS<m>:EBTB:DPLus:THReshold <ThresholdDplus>

Sets the high threshold (D +) of the signal, if `BUS<m>:EBTB:TYPE SINGle`.

Suffix:
<m> 1..4

Parameters:
<ThresholdDplus> Range: -10 to 10
Increment: 1E-3
*RST: 0
Default unit: V

BUS<m>:EBTB:FCSY <CustSync>

Sets a pattern value, if `BUS<m>:EBTB:SYNC CUS`.

Suffix:
<m> 1..4

Parameters:
<CustSync>

BUS<m>:EBTB:SCOunt?

Returns the symbol count for the selected serial bus, i.e. the number of symbols in the present acquisition.

Suffix:
<m> 1..4

Return values:
<Count> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:EBTB:SCSY <CustSync2>

Sets the value for the second pattern, if `BUS<m>:EBTB:SYNC CUS` and `BUS<m>:EBTB:USCS ON`.

Suffix:
<m> 1..4

Parameters:
<CustSync2>

BUS<m>:EBTB:SYNC <CommaSymbol>

Sets the sync symbol, a control symbol used for low level control functions.

Suffix:
<m> 1..4

Parameters:
<CommaSymbol> K285 | K281 | K287 | CUS
*RST: K285

BUS<m>:EBTB:TYPE <SignalType>

Selects the signal type that is used for the decoding. You can choose between a differential and single-ended signal.

Suffix:
<m> 1..4

Parameters:
<SignalType> DIFFerential | SINGle
*RST: DIFFerential

BUS<m>:EBTB:USCS <IsSecondSync>

Selects whether the second pattern is enabled, if `BUS<m>:EBTB:SYNC CUS`.

Suffix:
<m> 1..4

Parameters:
<IsSecondSync> ON | OFF
*RST: OFF

BUS<m>:EBTB:BITDetermi

Starts a software algorithm for the automatic determination of the bitrate.

Suffix:
<m> 1..4

Usage: Event
Asynchronous command

BUS<m>:EBTB:FAUToscale

Starts software algorithms for determining the signal threshold levels and bitrate.

Suffix:
<m> 1..4

Usage: Event
Asynchronous command

24.17.26.2 Trigger settings

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:EBTB:DISParityerr 2336
TRIGger<m>:EBTB:DX 2337
TRIGger<m>:EBTB:DY 2337
TRIGger<m>:EBTB:EBPA 2337
TRIGger<m>:EBTB:GLITcherror 2337
TRIGger<m>:EBTB:SSTYpe 2337
TRIGger<m>:EBTB:SYME 2338
TRIGger<m>:EBTB:SYMType 2338
TRIGger<m>:EBTB:TBPA 2338
TRIGger<m>:EBTB:STYPe 2338
TRIGger<m>:EBTB:UNK 2338

TRIGger<m>:EBTB:DISParityerr <DispError>

Defines, if a trigger on a disparity error is activated or not.

Parameters:
<DispError> ON | OFF
*RST: OFF

TRIGger<m>:EBTB:DX <DXSymbol>

Sets thex value of the data character Dx.y to be triggered on, if `TRIGger<m>:EBTB:SYMType` is set to `Dxy`.

Parameters:
<DXSymbol> Range: 0 to 31
Increment: 1
*RST: 0

TRIGger<m>:EBTB:DY <DYSymbol>

Sets the y value of the data character Dx.y to be triggered on, if `TRIGger<m>:EBTB:SYMType` is set to `Dxy`.

Parameters:
<DYSymbol> Range: 0 to 7
Increment: 1
*RST: 0

TRIGger<m>:EBTB:EBPA <EBPattern>

Sets the 8-bit pattern to be triggered on, if `TRIGger<m>:EBTB:STYPe` is set to `EB`.

Parameters:
<EBPattern>

TRIGger<m>:EBTB:GLITcherror <GlitchError>

Defines, if a trigger on a glitch error is activated or not.

Parameters:
<GlitchError> ON | OFF
*RST: OFF

TRIGger<m>:EBTB:SSTYpe <SymbolType>

Selects the symbol type to be triggered on. You can select a single symbol or an expression (defined series of symbols).

Parameters:
<SymbolType> SYMBol | EXPRession
*RST: SYMBol

TRIGger<m>:EBTB:SYME <Expression>

Selects the format of the symbol to be triggered on, if `TRIGger<m>:EBTB:SSTYpe` is set to `EXPRession`.

Parameters:
<Expression>

TRIGger<m>:EBTB:SYMType <SymbolType>

Selects the data character (Dx.y) or control character to bea triggered on. You can specify he value of the data character to be triggered on with `TRIGger<m>:EBTB:DX` and `TRIGger<m>:EBTB:DY`.

Parameters:
<SymbolType> DXY | K280 | K281 | K282 | K283 | K284 | K285 | K286 | K287 | K237 | K277 | K297 | K307

*RST: K285

TRIGger<m>:EBTB:TBPA <TBPattern>

Sets the 10-bit pattern to be triggered on, if `TRIGger<m>:EBTB:STYPe` is set to `TB`.

Parameters:
<TBPattern>

TRIGger<m>:EBTB:STYPe <Type>

Selects the type of condition to be triggered on for 8b10b.

Parameters:
<Type> KD | EB | TB | EXPRession | ERRor

`KD`: K/D symbol
`EB`: 8-bit pattern
`TB`: 10-bit pattern

*RST: KD

TRIGger<m>:EBTB:UNK <UnkError>

Defines, if a trigger on a unknown symbol error is activated or not.

Parameters:
<UnkError> ON | OFF

*RST: OFF

24.17.26.3 Decode results

In all `BUS<m>:EBTB:SYMBol<n>` commands, the suffix <m> selects the serial bus and the suffix <n> selects the symbol in the decode table.

BUS<m>:EBTB:SYMBol<n>:DATA?....2339
BUS<m>:EBTB:SYMBol<n>:STARt?....2339
BUS<m>:EBTB:SYMBol<n>:STATus?....2340
BUS<m>:EBTB:SYMBol<n>:STOP?....2340
BUS<m>:EBTB:SYMBol<n>:BYTE?....2340
BUS<m>:EBTB:SYMBol<n>:KDCode?....2341
BUS<m>:EBTB:SYMBol<n>:TENBit?....2341

BUS<m>:EBTB:SYMBol<n>:DATA?

Returns the data of the specified symbol.

The format is determined by the remote command `BUS<m>:EBTB:DISF`.

Suffix:
<m> 1..4

<n> *

Return values:
<Data>

Example:
```
BUS<m>:EBTB:DISF EB
BUS:EBTB:SYMBol5:DATA?
<-- BC+
```

Example:
```
BUS<m>:EBTB:DISF KD
BUS:EBTB:SYMBol5:DATA?
<-- K28.5+
```

Example:
```
BUS<m>:EBTB:DISF TB
BUS:EBTB:SYMBol5:DATA?
<-- 305
```

Usage: Query only

BUS<m>:EBTB:SYMBol<n>:STARt?

Returns the start time of the specified symbol.

Suffix:
<m> 1..4

<n> *

Return values:
<Start>

Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:EBTB:SYMBol<n>:STATus?

Returns the status of the specified symbol. It can show what kind of error was detected or no error.

Suffix:
<m> 1..4

<n> *

Return values:
<State> OK | UNK | DISPARITY | DISParity | GLITCH | GLITch

OK
There are no errors in the state of the specified symbol.

UNK
An unknown error was detected for the specified symbol.

DISPARITY = DISParity
A disparity error was detected for the specified symbol.

GLITCH = GLITch
A glitch error was detected for the specified symbol.

*RST: OK

Usage: Query only

BUS<m>:EBTB:SYMBol<n>:STOP?

Returns the stop time of the specified symbol.

Suffix:
<m> 1..4

<n> *

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:EBTB:SYMBol<n>:BYTE?

Returns the symbol byte value.

Suffix:
<m> 1..4

<n> *

Return values:

<Byte>	Range:	0 to 0
	Increment:	1
	*RST:	0

Usage: Query only

BUS<m>:EBTB:SYMBol<n>:KDCode?

Returns the K/D code.

Suffix:

<m>	1..4
<n>	*

Return values:
<KDCode>

Usage: Query only

BUS<m>:EBTB:SYMBol<n>:TENBit?

Returns the 10-bit symbol.

Suffix:

<m>	1..4
<n>	*

Return values:
<TenBit>

Usage: Query only

24.17.26.4 Search settings

SEARch:TRIGger:EBTB:DISParityerr....2341
SEARch:TRIGger:EBTB:DX....2342
SEARch:TRIGger:EBTB:DY....2342
SEARch:TRIGger:EBTB:EBPA....2342
SEARch:TRIGger:EBTB:GLITcherror....2342
SEARch:TRIGger:EBTB:STYPe....2343
SEARch:TRIGger:EBTB:SYME....2343
SEARch:TRIGger:EBTB:SYMType....2343
SEARch:TRIGger:EBTB:TBPA....2344
SEARch:TRIGger:EBTB:TYPE....2344
SEARch:TRIGger:EBTB:UNK....2344

SEARch:TRIGger:EBTB:DISParityerr <SearchName>,<DispError>
SEARch:TRIGger:EBTB:DISParityerr? <SearchName>

Defines, if a search for any disparity error is activated or not.

Parameters:
<DispError> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:EBTB:DX <SearchName>,<DXSymbol>
SEARch:TRIGger:EBTB:DX? <SearchName>

Sets the x value of the data character Dx.y to be searched for, if `SEARch:TRIGger:EBTB:SYMType` is set to `Dxy`.

Parameters:
<DXSymbol> Range: 0 to 31
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:EBTB:DY <SearchName>,<DYSymbol>
SEARch:TRIGger:EBTB:DY? <SearchName>

Sets the y value of the data character Dx.y to be searched for, if `SEARch:TRIGger:EBTB:SYMType` is set to `Dxy`.

Parameters:
<DYSymbol> Range: 0 to 7
Increment: 1
*RST: 0

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:EBTB:EBPA <SearchName>,<EBPattern>
SEARch:TRIGger:EBTB:EBPA? <SearchName>

Sets the 8-bit pattern to be searched for.

Parameters:
<EBPattern>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:EBTB:GLITcherror <SearchName>,<GlitchError>
SEARch:TRIGger:EBTB:GLITcherror? <SearchName>

Defines, if a search for any glitch error is activated or not.

Parameters:
<GlitchError> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:EBTB:STYPe <SearchName>,<Type>
SEARch:TRIGger:EBTB:STYPe? <SearchName>

Selects the search type.

Parameters:
<Type> KD | EB | TB | EXPRession | ERRor
`KD`: K/D symbol
`EB`: 8-bit pattern
`TB`: 10-bit pattern
*RST: KD

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:EBTB:SYME <SearchName>,<Expression>
SEARch:TRIGger:EBTB:SYME? <SearchName>

Selects the format of the symbol to be searched for, if `SEARch:TRIGger:EBTB:STYPe` is set to `EXPRession`.

Parameters:
<Expression>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:EBTB:SYMType <SearchName>,<SymbolType>
SEARch:TRIGger:EBTB:SYMType? <SearchName>

Selects the data character (Dx.y) or control character to bea searched for. You can specify he value of the data character to be searched for with `SEARch:TRIGger:EBTB:DX` and `SEARch:TRIGger:EBTB:DY`.

Parameters:
<SymbolType> DXY | K280 | K281 | K282 | K283 | K284 | K285 | K286 | K287 | K237 | K277 | K297 | K307
*RST: K285

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:EBTB:TBPA <SearchName>,<TBPattern>
SEARch:TRIGger:EBTB:TBPA? <SearchName>

Sets the 10-bit pattern to be searched for, if `SEARch:TRIGger:EBTB:STYPe` is set to `TB`.

Parameters:
<TBPattern>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:EBTB:TYPE <SearchName>,<Type>
SEARch:TRIGger:EBTB:TYPE? <SearchName>

Selects the type of condition to be searched for. You can search for a certain symbol or enable a specific error condition.

Parameters:
<Type> SYMBol | ERRor
*RST: SYMBol

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:EBTB:UNK <SearchName>,<UnkError>
SEARch:TRIGger:EBTB:UNK? <SearchName>

Defines, if a search for any unknown symbol error is activated or not.

Parameters:
<UnkError> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

24.17.26.5 Search results

In all `SEARch:RESult:EBTB:SYMBol<m>` commands, the suffix <m> selects the frame number in the list of search results.

SEARch:RESult:EBTB:SCOunt? 2345
SEARch:RESult:EBTB:SYMBol<m>:DATA? 2345
SEARch:RESult:EBTB:SYMBol<m>:STARt? 2345
SEARch:RESult:EBTB:SYMBol<m>:STATus? 2345
SEARch:RESult:EBTB:SYMBol<m>:STOP? 2346

SEARch:RESult:EBTB:SCOunt? <SearchName>

Returns the symbol count, i.e. the number of symbols found in the specified search result.

Query parameters:
<SearchName>

Return values:

<Count>	Range:	0 to 100000
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:EBTB:SYMBol<m>:DATA? <SearchName>

Returns the data of the specified symbol.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:EBTB:SYMBol<m>:STARt? <SearchName>

Returns the start time of the specified symbolwithin the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Start>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

SEARch:RESult:EBTB:SYMBol<m>:STATus? <SearchName>

Returns the status of the specified symbol within the search result. It can show what kind of error was detected or no error.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> OK | UNK | DISPARITY | DISParity | GLITCH | GLITch

OK
There are no errors in the state of the specified symbol.

UNK
An unknown error was detected for the specified symbol.

DISPARITY = DISParity
A disparity error was detected for the specified symbol.

GLITCH = GLITch
A glitch error was detected for the specified symbol.

*RST: OK

Usage: Query only

SEARch:RESult:EBTB:SYMBol<m>:STOP? <SearchName>

Returns the stop time of the specified symbol within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

24.17.27 PCIe (option R&S RTO6K72)

- Configuration....2346
- Clock data recovery....2349
- Trigger....2351
- Decode results....2359
- Search settings....2368
- Search results....2380

24.17.27.1 Configuration

In all `BUS<m>:PCIE` commands, the suffix <m> selects the serial bus.

BUS<m>:PCIE:DSCRambling 2347
BUS<m>:PCIE:FIDLe 2347
BUS<m>:PCIE:GEN 2347
BUS<m>:PCIE:LNKW 2348
BUS<m>:PCIE:LZER:SOURce 2348
BUS<m>:PCIE:LONE:SOURce 2348
BUS<m>:PCIE:LTWO:SOURce 2348
BUS<m>:PCIE:LTHRee:SOURce 2348
BUS<m>:PCIE:LZER:THRHigh 2348
BUS<m>:PCIE:LONE:THRHigh 2348
BUS<m>:PCIE:LTWO:THRHigh 2348
BUS<m>:PCIE:LTHRee:THRHigh 2348
BUS<m>:PCIE:LZER:THRLow 2349
BUS<m>:PCIE:LONE:THRLow 2349
BUS<m>:PCIE:LTWO:THRLow 2349
BUS<m>:PCIE:LTHRee:THRLow 2349
BUS<m>:PCIE:SMLanes 2349

BUS<m>:PCIE:DSCRambling <Active>

Enables descrambling of the data.

Suffix:
<m> 1..4

Parameters:
<Active> ON | OFF
*RST: ON

BUS<m>:PCIE:FIDLe <Active>

Enables the filtering of idle packets.

Suffix:
<m> 1..4

Parameters:
<Active> ON | OFF
*RST: ON

BUS<m>:PCIE:GEN <Generation>

Sets the generation of the PCIe technology.

Suffix:
<m> 1..4

Parameters:
<Generation> V1 | V2 | V3

V1
PCIe Gen1 (2.5 Gbit/s)

V2
PCIe Gen2 (5 Gbit/s)
*RST: V1

BUS<m>:PCIE:LNKW <LinkWidth>

Sets the link width, the number of lanes that are used for the transmission of the data.

Suffix:
<m> 1..4

Parameters:
<LinkWidth> X1 | X2 | X4
*RST: X1

BUS<m>:PCIE:LZER:SOURce <SourceL0>

Sets the signal sources for Lane 0.

Suffix:
<m> 1..4

Parameters:
<SourceL0> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4
*RST: C1W1

BUS<m>:PCIE:LONE:SOURce <SourceL1>
BUS<m>:PCIE:LTWO:SOURce <SourceL2>
BUS<m>:PCIE:LTHRee:SOURce <SourceL3>

Sets the signal sources for the corresponding logical lane.

Suffix:
<m> 1..4

Parameters:
<SourceL3> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | NONE
*RST: LONe: C2W1, LTWo: C3W1, LTHree: C4W1

BUS<m>:PCIE:LZER:THRHigh <ThresholdL0High>
BUS<m>:PCIE:LONE:THRHigh <ThresholdL1High>
BUS<m>:PCIE:LTWO:THRHigh <ThresholdL2High>
BUS<m>:PCIE:LTHRee:THRHigh <ThresholdL3High>

Sets the high threshold value for the respective lane.

Suffix:
<m> 1..4

Parameters:

<ThresholdL3High>	Range:	-5 to 5
	Increment:	1E-3
	*RST:	0.02
	Default unit:	V

BUS<m>:PCIE:LZER:THRLow <ThresholdL0Low>
BUS<m>:PCIE:LONE:THRLow <ThresholdL1Low>
BUS<m>:PCIE:LTWO:THRLow <ThresholdL2Low>
BUS<m>:PCIE:LTHRee:THRLow <ThresholdL3Low>
BUS<m>:PCIE:SMLanes <Active>

Enables the display of the result in multiple lanes.

Suffix:

<m>	1..4

Parameters:

<Active>	ON | OFF	
	*RST:	ON

24.17.27.2 Clock data recovery

BUS<m>:CDR:BITRate 2349
BUS<m>:CDR:PLL:BWIDth 2349
BUS<m>:CDR:PLL:DAMPing 2350
BUS<m>:CDR:PLL:ORDer 2350
BUS<m>:CDR:PLL:RELBwidth 2350
BUS<m>:CDR:RESults 2350
BUS<m>:CDR:SYNC 2351

BUS<m>:CDR:BITRate <Bitrate>

Sets the quiescent frequency of the PLL. It corresponds to the data rate of the data stream from which the clock is to be recovered.

Suffix:

<m>	1..4

Parameters:

<Bitrate>	Range:	100 to 5E+9
	Increment:	10
	*RST:	1E+9
	Default unit:	bps

BUS<m>:CDR:PLL:BWIDth <PLLBandwidth>

Sets the PLL bandwidth. It defines the part of the spectrum that the PLL can follow during synchronization. The PLL bandwidth is usually defined by the transmission standard.

Suffix:
<m> 1..4

Parameters:
<PLLBandwidth> Range: 20E+3 to 10E+6
Increment: 10
*RST: 599.88E+3
Default unit: Hz

BUS<m>:CDR:PLL:DAMPing <DampingFactor>

Sets the damping factor, which is only relevant for second order PLL.

Suffix:
<m> 1..4

Parameters:
<DampingFactor> Range: 0.5 to 1
Increment: 0.01
*RST: 0.7

BUS<m>:CDR:PLL:ORDer <PLLOrder>

Sets the order of the PLL: first or second order. PLL of higher order can compensate for more complex jitter behavior.

Suffix:
<m> 1..4

Parameters:
<PLLOrder> FIRSt | SECond
*RST: SECond

BUS<m>:CDR:PLL:RELBwidth <PLLRelBw>

Sets the relative bandwidth, that is the ratio of the nominal bit rate to the PLL bandwidth.

Suffix:
<m> 1..4

Parameters:
<PLLRelBw> Range: 10 to 5000
Increment: 1
*RST: 1667

BUS<m>:CDR:RESults <Results>

The PLL requires some time to synchronize to the phase of the data stream. You can select when the CDR algorithm returns clock edges.

Suffix:
<m> 1..4

Parameters:
<Results> ALL | AISYnc
*RST: ALL

BUS<m>:CDR:SYNC <InitialSync>

Defines the phase reference for the first clock edge.

Suffix:
<m> 1..4

Parameters:
<InitialSync> SAMPle | DATaedge

SAMPle
The first clock edge matches the first sample of the waveform at the left border of the display.

DATaedge
The first clock edge matches the first edge of the data signal.

*RST: DATaedge

24.17.27.3 Trigger

The trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on serial buses.

To trigger on a serial bus, make sure that:

- `TRIGger<m>:SOURce[:SELect]` is set to `SBUS`.
- The sources of the serial bus are channel signals: use `BUS<m>:...:SOURce` commands.
- Decoding is enabled: `BUS<m>[:STATe]` is set to `ON`.

TRIGger<m>:PCIE:DLLP:FCTL:CRET 2352
TRIGger<m>:PCIE:DLLP:FCTL:VCID 2352
TRIGger<m>:PCIE:DLLP:MULT:AUTH 2352
TRIGger<m>:PCIE:DLLP:MULT:CRET 2353
TRIGger<m>:PCIE:DLLP:MULT:DVPT 2353
TRIGger<m>:PCIE:DLLP:MULT:HABS 2353
TRIGger<m>:PCIE:DLLP:MULT:MIXT 2353
TRIGger<m>:PCIE:DLLP:MULT:PHAS 2353
TRIGger<m>:PCIE:DLLP:MULT:RESA 2353
TRIGger<m>:PCIE:DLLP:MULT:TLPT 2354
TRIGger<m>:PCIE:DLLP:MULT:TYPE 2354
TRIGger<m>:PCIE:DLLP:MULT:VHFC 2354
TRIGger<m>:PCIE:DLLP:MULT:VHGR 2354
TRIGger<m>:PCIE:DLLP:MULT:VLNR 2355
TRIGger<m>:PCIE:DLLP:POWM 2355
TRIGger<m>:PCIE:DLLP:SEQ 2355

TRIGger<m>:PCIE:DLLP:TYPE....2355
TRIGger<m>:PCIE:DLLP:VPAT....2355
TRIGger<m>:PCIE:ERRC:CRC....2355
TRIGger<m>:PCIE:ERRC:DISP....2356
TRIGger<m>:PCIE:ERRC:ECRC....2356
TRIGger<m>:PCIE:ERRC:INVP....2356
TRIGger<m>:PCIE:ERRC:LCRC....2356
TRIGger<m>:PCIE:OSET:TYPE....2356
TRIGger<m>:PCIE:TLP:ADRT....2357
TRIGger<m>:PCIE:TLP:CFGT....2357
TRIGger<m>:PCIE:TLP:CPID....2357
TRIGger<m>:PCIE:TLP:CPLS....2357
TRIGger<m>:PCIE:TLP:DEID....2357
TRIGger<m>:PCIE:TLP:MERW....2358
TRIGger<m>:PCIE:TLP:MSGC....2358
TRIGger<m>:PCIE:TLP:MSGR....2358
TRIGger<m>:PCIE:TLP:ORDE....2358
TRIGger<m>:PCIE:TLP:REID....2358
TRIGger<m>:PCIE:TLP:SNOO....2358
TRIGger<m>:PCIE:TLP:SNUM....2359
TRIGger<m>:PCIE:TLP:TCHN....2359
TRIGger<m>:PCIE:TLP:TYPE....2359
TRIGger<m>:PCIE:TYPE....2359

TRIGger<m>:PCIE:DLLP:FCTL:CRET <CreditType>

Sets credit type value to be triggered on for the data link layer trigger.

Parameters:

<CreditType> ANY | P | NP | CPL

*RST: ANY

TRIGger<m>:PCIE:DLLP:FCTL:VCID <VirtualChnID>

Sets the virtual channel ID to be triggered on for the data link layer trigger.

Parameters:

<VirtualChnID> ANY | ZERO | ONE | TWO | THREE | FOUR | FIVE | SIX | SEVEN

*RST: ANY

TRIGger<m>:PCIE:DLLP:MULT:AUTH <Authorized>

Sets the authorized presence state to be triggered on for the data link layer trigger.

Parameters:

<Authorized> ANY | YES | NO

*RST: ANY

TRIGger<m>:PCIE:DLLP:MULT:CRET <CreditType>

Sets the credit type to be triggered on for the multi-root data link layer trigger.

Parameters:

<CreditType> ANY | DATA | HEADER

*RST: ANY

TRIGger<m>:PCIE:DLLP:MULT:DVPT <DevicePortType>

Sets the device/port type to be triggered on for the data link layer trigger.

Parameters:

<DevicePortType> ANY | ZERO | ONE | TWO | THREE | FOUR | FIVE | SIX | SEVEN | EIGHT | NINE | TEN | ELEVEN | TWELVE | THIRTEEN | FOURTEEN | FIFTEEN

*RST: ANY

TRIGger<m>:PCIE:DLLP:MULT:HABS <VHAbsent>

Sets the state of the absent virtual hiearchies (VH) to be triggered on for the data link layer trigger.

Parameters:

<VHAbsent> ANY | YES | NO

*RST: ANY

TRIGger<m>:PCIE:DLLP:MULT:MIXT <MixedType>

Sets the mixed type presence state to be triggered on for the data link layer trigger.

Parameters:

<MixedType> ANY | YES | NO

*RST: ANY

TRIGger<m>:PCIE:DLLP:MULT:PHAS <Phase>

Sets the phase state to be triggered on for the data link layer trigger.

Parameters:

<Phase> ANY | YES | NO

*RST: ANY

TRIGger<m>:PCIE:DLLP:MULT:RESA <ResetA>

Sets the on the value of the A bit to be triggered on for the data link layer trigger.

Parameters:

<ResetA> ANY | ACK | REQUEST

*RST: ANY

TRIGger<m>:PCIE:DLLP:MULT:TLPT <TLPType>

Sets the multi-root type to be triggered on for the data link layer trigger.

Parameters:

<TLPType> ANY | P | NP | CPL

P
Posted credit.

NP
Non posted credit.

CL
Completion credit.

*RST: ANY

TRIGger<m>:PCIE:DLLP:MULT:TYPE <Type>

Sets the multi-root type for the data link layer to be triggered on.

Parameters:

<Type> MRINIT | MRRESET | MRUPDATEFC | MRINITFC1 | MRINITFC2

*RST: MRINIT

TRIGger<m>:PCIE:DLLP:MULT:VHFC <InitVHFC>

Sets the virtual hierarchies flow control presence state to be triggered on for the data link layer trigger.

Parameters:

<InitVHFC> ANY | YES | NO

*RST: ANY

TRIGger<m>:PCIE:DLLP:MULT:VHGR <ResetVHGroup>

Sets the virtual hierarchies group to be triggered on for the data link layer trigger.

Parameters:

<ResetVHGroup> ANY | ZERO | ONE | TWO | THREE | FOUR | FIVE | SIX | SEVEN | EIGHT | NINE | TEN | ELEVEN | TWELVE | THIRTEEN | FOURTEEN | FIFTEEN

*RST: ANY

TRIGger<m>:PCIE:DLLP:MULT:VLNR <VirtualLink>

Sets the virtual link (VL) number to be triggered on for the data link layer trigger.

Parameters:

<VirtualLink> ANY | ZERO | ONE | TWO | THREE | FOUR | FIVE | SIX | SEVEN

*RST: ANY

TRIGger<m>:PCIE:DLLP:POWM <PowerMngtType>

Sets the power management type to be triggered on for the data link layer trigger.

Parameters:

<PowerMngtType> ANY | ENTERL1 | ENTERL23 | ACTREQL1 | REQUESTACK

*RST: ANY

TRIGger<m>:PCIE:DLLP:SEQ <ACKNAKSequence>

Sets the sequence field, indicating what TLPs are affected by the ACK/NAK, to be triggered on for the data link layer trigger.

Parameters:

<ACKNAKSequence>

TRIGger<m>:PCIE:DLLP:TYPE <Type>

Sets the DLLP t type to be triggered on for the data link layer trigger.

Parameters:

<Type> ANY | MRDLLP | ACK | NAK | INITFC1 | INITFC2 | UPDATEFC | PM | VENDOR

*RST: ANY

TRIGger<m>:PCIE:DLLP:VPAT <VendorBtPatt>

Sets the vendor pattern to be triggered on for the data link layer trigger.

Parameters:

<VendorBtPatt> 24-bit pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:PCIE:ERRC:CRC <CRC16>

Enables triggering on 16-bit cyclic redundancy check (CRC) errors.

Parameters:

<CRC16> ON | OFF

*RST: OFF

TRIGger<m>:PCIE:ERRC:DISP <Disparity>

Enables triggering on disparity errors.

Parameters:

<Disparity> ON | OFF

*RST: OFF

TRIGger<m>:PCIE:ERRC:ECRC <ECRC>

Enables triggering on end-to-end cyclic redundancy checksum (ECRC) errors.

Parameters:

<ECRC> ON | OFF

*RST: OFF

TRIGger<m>:PCIE:ERRC:INVP <InvalidSymbol>

Enables triggering on invalid symbol errors.

Parameters:

<InvalidSymbol> ON | OFF

*RST: OFF

TRIGger<m>:PCIE:ERRC:LCRC <LCRC>

Enables triggering on link cyclic redundancy check (LCRC) errors.

Parameters:

<LCRC> ON | OFF

*RST: OFF

TRIGger<m>:PCIE:OSET:TYPE <Type>

Sets the ordered set to be triggered on.

Parameters:

<Type> SKP | TS1 | TS2 | FTS | EIOS | EIEOS | COMP

SKP
SKP ordered sets

TS1
Training sequence 1

TS2
Training sequence 2

FTS
Fast training sequence

EIOS
Electrical idle ordered set

EIEOS
Electrical idle exit ordered set

COMP
Compliance pattern

*RST: SKP

TRIGger<m>:PCIE:TLP:ADRT <AddressType>

Sets the address type to be triggered on for the transaction layer type trigger.

Parameters:

<AddressType> ANY | X32 | X64

*RST: ANY

TRIGger<m>:PCIE:TLP:CFGT <ConfigType>

Sets the configuration type to be triggered on for the transaction layer type trigger.

Parameters:

<ConfigType> ANY | TYPE0 | TYPE1

*RST: ANY

TRIGger<m>:PCIE:TLP:CPID <TLPCompleterID>

Sets the completer ID to be triggered on for the transaction layer type trigger.

Parameters:

<TLPCompleterID> 16-bit pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:PCIE:TLP:CPLS <CplStatus>

Sets the completion status to be triggered on for the transaction layer type trigger.

Parameters:

<CplStatus> ANY | SUCCESS | UNSUPPORTED | RETRY | ABORT

*RST: ANY

TRIGger<m>:PCIE:TLP:DEID <TLPDeviceID>

Sets the device ID to be triggered on for the transaction layer type trigger.

Parameters:

<TLPDeviceID> 16-bit pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:PCIE:TLP:MERW <MemRW>

Sets the read/ write type to be triggered on for the transaction layer type trigger.

Parameters:

<MemRW> ANY | READ | WRITE

*RST: ANY

TRIGger<m>:PCIE:TLP:MSGC <TLPMessageCode>

Sets the message code to be triggered on for the transaction layer type trigger.

Parameters:

<TLPMessageCode> 8-bit pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:PCIE:TLP:MSGR <MsgRouting>

Sets the message routing type to be triggered on for the transaction layer type trigger.

Parameters:

<MsgRouting> ANY | ROOTCOMPLEX | ADDRESS | DEVICEID | DEViceid | BROADCAST | LOCAL | GATHERED

DEVICEID = DEViceid

*RST: ANY

TRIGger<m>:PCIE:TLP:ORDE <Ordering>

Sets the ordering type to be triggered on for the transaction layer type trigger.

Parameters:

<Ordering> ANY | STRONG | RELAX

*RST: ANY

TRIGger<m>:PCIE:TLP:REID <TLPRequesterID>

Sets the requester ID to be triggered on for the transaction layer type trigger.

Parameters:

<TLPRequesterID> 16-bit pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:PCIE:TLP:SNOO <Snoop>

Sets the snoop state to be triggered on for the transaction layer type trigger.

Parameters:
<Snoop> ANY | YES | NO
*RST: ANY

TRIGger<m>:PCIE:TLP:SNUM <TLPSeqNo>

Sets the sequence numbe to be triggered on for the transaction layer type trigger.

Parameters:
<TLPSeqNo> 12-bit pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

TRIGger<m>:PCIE:TLP:TCHN <TrafficClass>

Sets the traffic class to be triggered on for the transaction layer type trigger.

Parameters:
<TrafficClass> ANY | ZERO | ONE | TWO | THREE | FOUR | FIVE | SIX | SEVEN
*RST: ANY

TRIGger<m>:PCIE:TLP:TYPE <Type>

Sets the transaction type to be triggered on for the transaction layer type trigger.

Parameters:
<Type> ANY | MEM | IO | CFG | MSG | CPL | FETCH | SWAP | CAS
*RST: ANY

TRIGger<m>:PCIE:TYPE <Type>

Sets the trigger type for the PCI express analysis..

Parameters:
<Type> TLP | DLLP | ORD | ERR
*RST: TLP

24.17.27.4 Decode results

In all `BUS<m>:PCIE:RESult:PACKet<n>` commands, the suffix <m> selects the serial bus and the suffix <n> selects the frame number in the decode table.

BUS<m>:PCIE:RESult:PACKet<n>:ADDR? 2360
BUS<m>:PCIE:RESult:PACKet<n>:CPID? 2360
BUS<m>:PCIE:RESult:PACKet<n>:CPS? 2360
BUS<m>:PCIE:RESult:PACKet<n>:DATA? 2361
BUS<m>:PCIE:RESult:PACKet<n>:DFC? 2361
BUS<m>:PCIE:RESult:PACKet<n>:FCOunt? 2362

BUS<m>:PCIE:RESult:PACKet<n>:FLD<o>:FVAL?....2362
BUS<m>:PCIE:RESult:PACKet<n>:FLD<o>:STATus?.... 2362
BUS<m>:PCIE:RESult:PACKet<n>:FLD<o>:TYPE?.... 2363
BUS<m>:PCIE:RESult:PACKet<n>:FLD<o>:VAL?....2363
BUS<m>:PCIE:RESult:PACKet<n>:HFC?....2364
BUS<m>:PCIE:RESult:PACKet<n>:LEN?....2364
BUS<m>:PCIE:RESult:PACKet<n>:RQID?.... 2364
BUS<m>:PCIE:RESult:PACKet<n>:SEQN?....2364
BUS<m>:PCIE:RESult:PACKet<n>:STARt?....2365
BUS<m>:PCIE:RESult:PACKet<n>:STATe?....2365
BUS<m>:PCIE:RESult:PACKet<n>:STOP?.... 2366
BUS<m>:PCIE:RESult:PACKet<n>:TAG?....2366
BUS<m>:PCIE:RESult:PACKet<n>:TYPE?....2366
BUS<m>:PCIE:RESult:PCOunt?.... 2367
BUS<m>:PCIE:RESult:PACKet<n>:BITRate?.... 2367

BUS<m>:PCIE:RESult:PACKet<n>:ADDR?

Returns the packet address for the specified packet.

Suffix:

<m>	1..4
<n>	*

Return values:

<Address>	Range:	0 to 0
	Increment:	1
	*RST:	0

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:CPID?

Returns the completer ID of the specified packet.

Suffix:

<m>	1..4
<n>	*

Return values:

<CompleterID>	Range:	0 to 65535
	Increment:	1
	*RST:	0

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:CPS?

Returns the completion status field value of the specified packet.

Suffix:
<m> 1..4

<n> *

Return values:
<CplStatus> RESERVED | SC | UR | CRS | CA

RESERVED
Others

SC
Successful Completion

UR
Unsupported Request

CRS
Configuration Request Retry Status

CA
Completer Abort

*RST: RESERVED

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:DATA?

Returns the data value of the specified packet.

Suffix:
<m> 1..4

<n> *

Return values:
<Data>

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:DFC?

Returns the credit value of the payload data of the specified packet.

Suffix:
<m> 1..4

<n> *

Return values:
<DataFC> Range: 0 to 4095
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:FCOunt?

Returns the number of decoded fields in the selected packet.

Suffix:

<m> 1..4

<n> *

Return values:

<FieldCount> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:FLD<o>:FVAL?

Returns the formatted value of the field with the selected field number within the frame with the selected packet number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

<o> *
Selects the field number.

Return values:

<ValueFormatted> String parameter

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:FLD<o>:STATus?

Returns the status of the frame with the selected field number within the packet with the selected packet number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

<o> *
Selects the field number.

Return values:

<State> OK | MALF | CRC16 | ECRC | LCRC | LEN | POE | MEE | UNCorrelated | FCRC | FPERror | UNKNown | OUTRange | INComplete | SYNCerror

MALF
Malformed packet

CRC16
CRC16 error

ECRC
ECRC error

LCRC
Link CRC error

LEN
Length error

POE
TLP prefix error

MEE
Max end-end TLP prefix error

*RST: OK

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:FLD<o>:TYPE?

Returns the type of the field with the selected field number within the frame with the selected packet number (for the selected serial bus).

Suffix:
<m> 1..4

<n> *

<o> *
Selects the field number.

Return values:
<Type>

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:FLD<o>:VAL?

Returns the value of the field with the selected field number within the packet with the selected packet number (for the selected serial bus).

Suffix:
<m> 1..4

<n> *

<o> *
Selects the field number.

Return values:
<FieldValue> Range: 0 to 0
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:HFC?

Returns the credit value of the header of the specified packet.

Suffix:

<m>	1..4
<n>	*

Return values:

<HeaderFC>	Range:	0 to 255
	Increment:	1
	*RST:	0

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:LEN?

Returns the length of the specified packet.

Suffix:

<m>	1..4
<n>	*

Return values:

<Length>	Range:	0 to 1023
	Increment:	1
	*RST:	0

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:RQID?

Returns the requester ID of the specified packet.

Suffix:

<m>	1..4
<n>	*

Return values:

<RequesterID>	Range:	0 to 65535
	Increment:	1
	*RST:	0

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:SEQN?

Returns the sequence number of the specified packet.

Suffix:

<m>	1..4
<n>	*

Return values:

<SequenceNumber> Range: 0 to 4095
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:STARt?

Returns the start time of the specified packet.

Suffix:

<m> 1..4

<n> *

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:STATe?

Returns the state of the specified packet.

Suffix:

<m> 1..4

<n> *

Return values:

<State> OK | MALF | CRC16 | ECRC | LCRC | LEN | POE | MEE | UNCorrelated | FCRC | FPERror | UNKNown | OUTRange | INComplete | SYNCerror

MALF
Mal formated packet

CRC16
CRC16 error

ECRC
ECRC error

LCRC
LCRC error

LEN
Length error

POE
TLP prefix error

MEE
Max end-end TLP prefix error

*RST: OK

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:STOP?

Returns the stop time of the specified packet.

Suffix:
<m> 1..4

<n> *

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:TAG?

Returns the tag of the specified packet.

Suffix:
<m> 1..4

<n> *

Return values:
<Tag> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:TYPE?

Returns the type of the specified packet. For a description of the packet types, see Chapter 13.26.1, "The PCIe protocol", on page 935.

Suffix:
<m> 1..4

<n> *

Return values:

<Type> MRD32 | MRD64 | MRDLK32 | MRDLK64 | MWR32 | MWR64 | IORD | IOWR | CFGRD0 | CFGWR0 | CFGRD1 | CFGWR1 | MSG | MSGD | CPL | CPLD | CPLLK | CPLDLK | FA32 | FA64 | SWP32 | SWP64 | CAS32 | CAS64 | LPRFX | EPRFX | ACK | NAK | PMEL1 | PMEL23 | PMASRL1 | PMRA | VENDS | IFC1P | IFC1NP | IFC1CPL | IFC2P | IFC2NP | IFC2CPL | UPDFCP | UPDFCNP | UPDFCCPL | MRUPDFC | MRIFC1 | MRIFC2 | MRINIT | MRRESET | UNK | SKPOS | EIOS | FTSOS | TS1OS | TS2OS | EIEOS | COMPL | MCOMPL | DCOMPL | UDEF | IDLE | SDS | EDB | EDS | SDAT | SCTR | PMUX | UNKTlp | UNKDllp | SINV

*RST: UNK

Usage: Query only

BUS<m>:PCIE:RESult:PCOunt?

Returns the packet count for the selected serial bus, i.e. the number of packets in the present acquisition.

Suffix:

<m> 1..4

Return values:

<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:PCIE:RESult:PACKet<n>:BITRate?

Returns the bit rate of the selected packet.

Suffix:

<m> 1..4

<n> *

Return values:

<PrimaryBitRate> Range: 0 to 100000000000
Increment: 1
*RST: 0
Default unit: bps

Usage: Query only

24.17.27.5 Search settings

SEARch:TRIGger:PCIE:DLLP:FCTL:CRET....2369
SEARch:TRIGger:PCIE:DLLP:FCTL:VCID....2369
SEARch:TRIGger:PCIE:DLLP:MULT:AUTH....2369
SEARch:TRIGger:PCIE:DLLP:MULT:CRET....2369
SEARch:TRIGger:PCIE:DLLP:MULT:DVPT....2370
SEARch:TRIGger:PCIE:DLLP:MULT:HABS....2370
SEARch:TRIGger:PCIE:DLLP:MULT:MIXT....2370
SEARch:TRIGger:PCIE:DLLP:MULT:PHAS....2370
SEARch:TRIGger:PCIE:DLLP:MULT:RESA....2371
SEARch:TRIGger:PCIE:DLLP:MULT:TLPT....2371
SEARch:TRIGger:PCIE:DLLP:MULT:TYPE....2371
SEARch:TRIGger:PCIE:DLLP:MULT:VHFC....2371
SEARch:TRIGger:PCIE:DLLP:MULT:VHGR....2372
SEARch:TRIGger:PCIE:DLLP:MULT:VLNR....2372
SEARch:TRIGger:PCIE:DLLP:POWM....2372
SEARch:TRIGger:PCIE:DLLP:SEQ....2373
SEARch:TRIGger:PCIE:DLLP:TYPE....2373
SEARch:TRIGger:PCIE:DLLP:VPAT....2373
SEARch:TRIGger:PCIE:ERRC:CRC....2373
SEARch:TRIGger:PCIE:ERRC:DISP....2374
SEARch:TRIGger:PCIE:ERRC:ECRC....2374
SEARch:TRIGger:PCIE:ERRC:INVP....2374
SEARch:TRIGger:PCIE:ERRC:LCRC....2374
SEARch:TRIGger:PCIE:OSET:COMP....2375
SEARch:TRIGger:PCIE:OSET:EIDE....2375
SEARch:TRIGger:PCIE:OSET:EIDL....2375
SEARch:TRIGger:PCIE:OSET:FTS....2375
SEARch:TRIGger:PCIE:OSET:SKIP....2376
SEARch:TRIGger:PCIE:OSET:TSONe....2376
SEARch:TRIGger:PCIE:OSET:TSTWo....2376
SEARch:TRIGger:PCIE:TLP:ADRT....2376
SEARch:TRIGger:PCIE:TLP:CFGT....2377
SEARch:TRIGger:PCIE:TLP:CPID....2377
SEARch:TRIGger:PCIE:TLP:CPLS....2377
SEARch:TRIGger:PCIE:TLP:DEID....2377
SEARch:TRIGger:PCIE:TLP:MERW....2378
SEARch:TRIGger:PCIE:TLP:MSGC....2378
SEARch:TRIGger:PCIE:TLP:MSGR....2378
SEARch:TRIGger:PCIE:TLP:ORDE....2378
SEARch:TRIGger:PCIE:TLP:REID....2379
SEARch:TRIGger:PCIE:TLP:SNOO....2379
SEARch:TRIGger:PCIE:TLP:SNUM....2379
SEARch:TRIGger:PCIE:TLP:TCHN....2379
SEARch:TRIGger:PCIE:TLP:TYPE....2380
SEARch:TRIGger:PCIE:TYPE....2380

SEARch:TRIGger:PCIE:DLLP:FCTL:CRET <SearchName>,<CreditType>
SEARch:TRIGger:PCIE:DLLP:FCTL:CRET? <SearchName>

Sets credit type value to be searched for, for the data link layer.

Parameters:
<CreditType> ANY | P | NP | CPL
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:FCTL:VCID <SearchName>,<VirtualChnID>
SEARch:TRIGger:PCIE:DLLP:FCTL:VCID? <SearchName>

Sets the virtual channel ID to be searched for, for the data link layer.

Parameters:
<VirtualChnID> ANY | ZERO | ONE | TWO | THREE | FOUR | FIVE | SIX | SEVEN
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:MULT:AUTH <SearchName>,<Authorized>
SEARch:TRIGger:PCIE:DLLP:MULT:AUTH? <SearchName>

Sets the authorized presence state to be searched for, for the data link layer.

Parameters:
<Authorized> ANY | YES | NO
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:MULT:CRET <SearchName>,<CreditType>
SEARch:TRIGger:PCIE:DLLP:MULT:CRET? <SearchName>

Sets the credit type to be searched for, for the data link layer.

Parameters:
<CreditType> ANY | DATA | HEADER
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:MULT:DVPT <SearchName>,<DevicePortType>
SEARch:TRIGger:PCIE:DLLP:MULT:DVPT? <SearchName>

Sets the device/port type to be searched for, for the data link layer.

Parameters:
<DevicePortType> ANY | ZERO | ONE | TWO | THREE | FOUR | FIVE | SIX | SEVEN | EIGHT | NINE | TEN | ELEVEN | TWELVE | THIRTEEN | FOURTEEN | FIFTEEN

*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:MULT:HABS <SearchName>,<VHAbsent>
SEARch:TRIGger:PCIE:DLLP:MULT:HABS? <SearchName>

Sets the state of the absent virtual hieararchies (VH) to be searched for, for the data link layer.

Parameters:
<VHAbsent> ANY | YES | NO

*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:MULT:MIXT <SearchName>,<MixedType>
SEARch:TRIGger:PCIE:DLLP:MULT:MIXT? <SearchName>

Sets the mixed type presence state to be searched for, for the data link layer.

Parameters:
<MixedType> ANY | YES | NO

*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:MULT:PHAS <SearchName>,<Phase>
SEARch:TRIGger:PCIE:DLLP:MULT:PHAS? <SearchName>

Sets the phase state to be searched for, for the data link layer.

Parameters:
<Phase> ANY | YES | NO

*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:MULT:RESA <SearchName>,<ResetA>
SEARch:TRIGger:PCIE:DLLP:MULT:RESA? <SearchName>

Sets the on the value of the A bit to be searched for, for the data link layer.

Parameters:
<ResetA> ANY | ACK | REQUEST
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:MULT:TLPT <SearchName>,<Type>
SEARch:TRIGger:PCIE:DLLP:MULT:TLPT? <SearchName>

Sets the multi-root type to be searched for, for the data link layer.

Parameters:
<Type> ANY | P | NP | CPL

P
Posted credit.

NP
Non posted credit.

CL
Completion credit.

*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:MULT:TYPE <SearchName>,<MRType>
SEARch:TRIGger:PCIE:DLLP:MULT:TYPE? <SearchName>

Sets the multi-root type to be searched for, for the data link layer.

Parameters:
<MRType> MRINIT | MRRESET | MRUPDATEFC | MRINITFC1 | MRINITFC2
*RST: MRINIT

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:MULT:VHFC <SearchName>,<InitVHFC>
SEARch:TRIGger:PCIE:DLLP:MULT:VHFC? <SearchName>

Sets the virtual hierarchies flow control presence state to be searched for, for the data link layer.

Parameters:
<InitVHFC> ANY | YES | NO
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:MULT:VHGR <SearchName>,<ResetVHGroup>
SEARch:TRIGger:PCIE:DLLP:MULT:VHGR? <SearchName>

Sets the virtual hierarchies group to be searched for, for the data link layer.

Parameters:
<ResetVHGroup> ANY | ZERO | ONE | TWO | THREE | FOUR | FIVE | SIX | SEVEN | EIGHT | NINE | TEN | ELEVEN | TWELVE | THIRTEEN | FOURTEEN | FIFTEEN
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:MULT:VLNR <SearchName>,<VirtualLink>
SEARch:TRIGger:PCIE:DLLP:MULT:VLNR? <SearchName>

Sets the virtual link (VL) number to be searched for, for the data link layer.

Parameters:
<VirtualLink> ANY | ZERO | ONE | TWO | THREE | FOUR | FIVE | SIX | SEVEN
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:POWM <SearchName>,<PowerMngtType>
SEARch:TRIGger:PCIE:DLLP:POWM? <SearchName>

Sets the power management type to be searched for, for the data link layer.

Parameters:
<PowerMngtType> ANY | ENTERL1 | ENTERL23 | ACTREQL1 | REQUESTACK
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:SEQ <SearchName>,<ACKNAKSequence>
SEARch:TRIGger:PCIE:DLLP:SEQ? <SearchName>

Sets the sequence field, indicating what TLPs are affected by the ACK/NAK, to be searched for, for the data link layer.

Parameters:
<ACKNAKSequence>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:TYPE <SearchName>,<Type>
SEARch:TRIGger:PCIE:DLLP:TYPE? <SearchName>

Sets the DLLP t type to be searched for, for the data link layer.

Parameters:
<Type> ANY | MRDLLP | ACK | NAK | INITFC1 | INITFC2 | UPDATEFC | PM | VENDOR

*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:DLLP:VPAT <SearchName>,<VendorBtPatt>
SEARch:TRIGger:PCIE:DLLP:VPAT? <SearchName>

Sets the vendor pattern to be searched for, for the data link layer.

Parameters:
<VendorBtPatt> 24-bit pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:ERRC:CRC <SearchName>,<CRC16>
SEARch:TRIGger:PCIE:ERRC:CRC? <SearchName>

Enables searching for 16-bit cyclic redundancy check (CRC) errors.

Parameters:
<CRC16> ON | OFF

*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:ERRC:DISP <SearchName>,<Disparity>
SEARch:TRIGger:PCIE:ERRC:DISP? <SearchName>

Enables searching for disparity errors.

Parameters:
<Disparity> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:ERRC:ECRC <SearchName>,<ECRC>
SEARch:TRIGger:PCIE:ERRC:ECRC? <SearchName>

Enables searching for end-to-end cyclic redundancy checksum (ECRC) errors.

Parameters:
<ECRC> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:ERRC:INVP <SearchName>,<InvalidSymbol>
SEARch:TRIGger:PCIE:ERRC:INVP? <SearchName>

Enables searching for invalid symbol errors.

Parameters:
<InvalidSymbol> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:ERRC:LCRC <SearchName>,<LCRC>
SEARch:TRIGger:PCIE:ERRC:LCRC? <SearchName>

Enables searching for link cyclic redundancy check (LCRC) errors.

Parameters:
<LCRC> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:OSET:COMP <SearchName>,<Format>
SEARch:TRIGger:PCIE:OSET:COMP? <SearchName>

Enables searching forcompliance pattern ordered sets.

Parameters:
<Format> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:OSET:EIDE <SearchName>,<Format>
SEARch:TRIGger:PCIE:OSET:EIDE? <SearchName>

Enables searching for electrical idle exit ordered sets.

Parameters:
<Format> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:OSET:EIDL <SearchName>,<Format>
SEARch:TRIGger:PCIE:OSET:EIDL? <SearchName>

Enables searching for electrical idle ordered sets.

Parameters:
<Format> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:OSET:FTS <SearchName>,<Format>
SEARch:TRIGger:PCIE:OSET:FTS? <SearchName>

Enables searching for fast training sequence ordered sets.

Parameters:
<Format> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:OSET:SKIP <SearchName>,<Format>
SEARch:TRIGger:PCIE:OSET:SKIP? <SearchName>

Enables searching for SKP ordered sets.

Parameters:
<Format> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:OSET:TSONe <SearchName>,<Format>
SEARch:TRIGger:PCIE:OSET:TSONe? <SearchName>

Enables searching for training sequence 1 ordered sets.

Parameters:
<Format> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:OSET:TSTWo <SearchName>,<Format>
SEARch:TRIGger:PCIE:OSET:TSTWo? <SearchName>

Enables searching for training sequence 2 ordered sets.

Parameters:
<Format> ON | OFF
*RST: OFF

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:TLP:ADRT <SearchName>,<AddressType>
SEARch:TRIGger:PCIE:TLP:ADRT? <SearchName>

Sets the address type to be searched for in the transaction layer type.

Parameters:
<AddressType> ANY | X32 | X64
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:TLP:CFGT <SearchName>,<CfgType>
SEARch:TRIGger:PCIE:TLP:CFGT? <SearchName>

Sets the configuration type to be searched for in the transaction layer type.

Parameters:
<CfgType> ANY | TYPE0 | TYPE1
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:TLP:CPID <SearchName>,<TLPCompleterID>
SEARch:TRIGger:PCIE:TLP:CPID? <SearchName>

Sets the completer ID to be searched for in the transaction layer type.

Parameters:
<TLPCompleterID> 16-bit pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:TLP:CPLS <SearchName>,<CplStatus>
SEARch:TRIGger:PCIE:TLP:CPLS? <SearchName>

Sets the completion status to be searched for in the transaction layer type.

Parameters:
<CplStatus> ANY | SUCCESS | UNSUPPORTED | RETRY | ABORT
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:TLP:DEID <SearchName>,<TLPDeviceID>
SEARch:TRIGger:PCIE:TLP:DEID? <SearchName>

Sets the device ID to be searched for in the transaction layer type.

Parameters:
<TLPDeviceID> 16-bit pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:TLP:MERW <SearchName>,<MemRW>
SEARch:TRIGger:PCIE:TLP:MERW? <SearchName>

Sets the read/ write type to be searched for in the transaction layer type.

Parameters:
<MemRW> ANY | READ | WRITE
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:TLP:MSGC <SearchName>,<TLPMessageCode>
SEARch:TRIGger:PCIE:TLP:MSGC? <SearchName>

Sets the message code to be searched for in the transaction layer type.

Parameters:
<TLPMessageCode> 8-bit pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:TLP:MSGR <SearchName>,<MsgRouting>
SEARch:TRIGger:PCIE:TLP:MSGR? <SearchName>

Sets the message routing type to be searched for in the transaction layer type.

Parameters:
<MsgRouting> ANY | ROOTCOMPLEX | ADDRESS | DEVICEID | DEViceid | BROADCAST | LOCAL | GATHERED
DEVICEID = DEViceid
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:TLP:ORDE <SearchName>,<Ordering>
SEARch:TRIGger:PCIE:TLP:ORDE? <SearchName>

Sets the ordering type to be searched for in the transaction layer type.

Parameters:
<Ordering> ANY | STRONG | RELAX
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:TLP:REID <SearchName>,<TLPRequesterID>
SEARch:TRIGger:PCIE:TLP:REID? <SearchName>

Sets the requester ID to be searched for in the transaction layer type.

Parameters:

<TLPRequesterID> 16-bit pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:PCIE:TLP:SNOO <SearchName>,<Snoop>
SEARch:TRIGger:PCIE:TLP:SNOO? <SearchName>

Sets the snoop state to be searched for in the transaction layer type.

Parameters:

<Snoop> ANY | YES | NO

*RST: ANY

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:PCIE:TLP:SNUM <SearchName>,<TLPSeqNo>
SEARch:TRIGger:PCIE:TLP:SNUM? <SearchName>

Sets the sequence numbe to be searched for in the transaction layer type.

Parameters:

<TLPSeqNo> 12-bit pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The string parameter accepts the bit value X (don't care).

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:PCIE:TLP:TCHN <SearchName>,<TrafficClass>
SEARch:TRIGger:PCIE:TLP:TCHN? <SearchName>

Sets the traffic class to be searched for in the transaction layer type.

Parameters:

<TrafficClass> ANY | ZERO | ONE | TWO | THREE | FOUR | FIVE | SIX | SEVEN

*RST: ANY

Parameters for setting and query:

<SearchName>

SEARch:TRIGger:PCIE:TLP:TYPE <SearchName>,<Type>
SEARch:TRIGger:PCIE:TLP:TYPE? <SearchName>

Sets the transaction type to be searched for in the transaction layer type.

Parameters:
<Type> ANY | MEM | IO | CFG | MSG | CPL | FETCH | SWAP | CAS
*RST: ANY

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:PCIE:TYPE <SearchName>,<Type>
SEARch:TRIGger:PCIE:TYPE? <SearchName>

Parameters:
<Type> TLP | DLLP | ORD | ERR
*RST: TLP

Parameters for setting and query:
<SearchName>

24.17.27.6 Search results

In all `SEARch:RESult:PCIE:PACKet<m>` commands, the suffix <m> selects the frame number in the list of search results.

In search result commands, you have to specify the `<SearchName>` parameter. It is a string parameter that contains the search definition name.

For a description of the returned values, see the corresponding commands in Chapter 24.17.27.4, "Decode results", on page 2359.

SEARch:RESult:PCIE:PACKet<m>:ADDR? 2381
SEARch:RESult:PCIE:PACKet<m>:CPID? 2381
SEARch:RESult:PCIE:PACKet<m>:CPS? 2381
SEARch:RESult:PCIE:PACKet<m>:DATA? 2381
SEARch:RESult:PCIE:PACKet<m>:DFC? 2382
SEARch:RESult:PCIE:PACKet<m>:FCOunt? 2382
SEARch:RESult:PCIE:PACKet<m>:FLD<n>:FVAL? 2382
SEARch:RESult:PCIE:PACKet<m>:FLD<n>:STATus? 2382
SEARch:RESult:PCIE:PACKet<m>:FLD<n>:TYPE? 2383
SEARch:RESult:PCIE:PACKet<m>:FLD<n>:VAL? 2383
SEARch:RESult:PCIE:PACKet<m>:HFC? 2383
SEARch:RESult:PCIE:PACKet<m>:LEN? 2384
SEARch:RESult:PCIE:PACKet<m>:RQID? 2384
SEARch:RESult:PCIE:PACKet<m>:SEQN? 2384
SEARch:RESult:PCIE:PACKet<m>:STARt? 2385
SEARch:RESult:PCIE:PACKet<m>:STATe? 2385
SEARch:RESult:PCIE:PACKet<m>:STOP? 2385

SEARch:RESult:PCIE:PACKet<m>:TAG?.....2385
SEARch:RESult:PCIE:PACKet<m>:TYPE?.....2386
SEARch:RESult:PCIE:PCOunt?..... 2386

SEARch:RESult:PCIE:PACKet<m>:ADDR? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Address>

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:CPID? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<CompleterID> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:CPS? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<CompleteState> RESERVED | SC | UR | CRS | CA
*RST: RESERVED

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:DATA? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Data>

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:DFC? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<DataFC> Range: 0 to 4095
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:FCOunt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<FieldCount> Range: 0 to 65535
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:FLD<n>:FVAL? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName> String parameter

Return values:
<ValueFormatted> String parameter

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:FLD<n>:STATus? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<State> OK | MALF | CRC16 | ECRC | LCRC | LEN | POE | MEE | UNCorrelated | FCRC | FPERror | UNKNown | OUTRange | INComplete | SYNCerror

*RST: OK

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:FLD<n>:TYPE? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<Type>

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:FLD<n>:VAL? <SearchName>

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<FieldValue> Range: 0 to 0
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:HFC? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<HeaderFC>	Range:	0 to 255
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:LEN? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Length>	Range:	0 to 1023
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:RQID? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<RequesterID>	Range:	0 to 65535
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:SEQN? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<SequenceNumber>	Range:	0 to 4095
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:STARt? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:STATe? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<State> OK | MALF | CRC16 | ECRC | LCRC | LEN | POE | MEE | UNCorrelated | FCRC | FPERror | UNKNown | OUTRange | INComplete | SYNCerror
*RST: OK

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:STOP? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:TAG? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Tag> Range: 0 to 255
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:PCIE:PACKet<m>:TYPE? <SearchName>

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:
<Type> MRD32 | MRD64 | MRDLK32 | MRDLK64 | MWR32 | MWR64 | IORD | IOWR | CFGRD0 | CFGWR0 | CFGRD1 | CFGWR1 | MSG | MSGD | CPL | CPLD | CPLLK | CPLDLK | FA32 | FA64 | SWP32 | SWP64 | CAS32 | CAS64 | LPRFX | EPRFX | ACK | NAK | PMEL1 | PMEL23 | PMASRL1 | PMRA | VENDS | IFC1P | IFC1NP | IFC1CPL | IFC2P | IFC2NP | IFC2CPL | UPDFCP | UPDFCNP | UPDFCCPL | MRUPDFC | MRIFC1 | MRIFC2 | MRINIT | MRRESET | UNK | SKPOS | EIOS | FTSOS | TS1OS | TS2OS | EIEOS | COMPL | MCOMPL | DCOMPL | UDEF | IDLE | SDS | EDB | EDS | SDAT | SCTR | PMUX | UNKTlp | UNKDllp | SINV
*RST: UNK

Usage: Query only

SEARch:RESult:PCIE:PCOunt? <SearchName>

Query parameters:
<SearchName>

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

24.17.28 DDR (option R&S RTO6-K91)

- Configuration....2387
- Decode results....2389
- Search settings....2393
- Search results....2398
- Eye diagram analysis....2402

24.17.28.1 Configuration

In all `BUS<m>:DDR` commands, the suffix <m> selects the serial bus.

BUS<m>:DDR:STANdard....2387
BUS<m>:DDR:DATA:HYSTeresis....2387
BUS<m>:DDR:DATA:SOURce....2388
BUS<m>:DDR:DATA:THReshold....2388
BUS<m>:DDR:STRBe:HYSTeresis....2388
BUS<m>:DDR:STRBe:SOURce....2388
BUS<m>:DDR:STRBe:THReshold....2389
BUS<m>:DDR:BITRate....2389

BUS<m>:DDR:STANdard <Standard>

Selects the DDR standard.

Suffix:
<m> 1..4

Parameters:
<Standard> DDR3

`DDR3`: requires option R&S RTO6-K91.

*RST: DDR3

Usage: Asynchronous command

BUS<m>:DDR:DATA:HYSTeresis <DataHysteresis>

Sets a value for the hysteresis of the data.

Suffix:
<m> 1..4

Parameters:
<DataHysteresis> Range: 0 to 10
Increment: 0.01
*RST: 0.15
Default unit: V

Usage: Asynchronous command

BUS<m>:DDR:DATA:SOURce <DataSource>

Selects the source for the data output signal.

Suffix:
<m> 1..4

Parameters:
<DataSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

*RST: C1W1

Usage: Asynchronous command

BUS<m>:DDR:DATA:THReshold <DataThreshold>

Sets the threshold value for the digitization of the data line.

Suffix:
<m> 1..4

Parameters:
<DataThreshold> Range: -10 to 10
Increment: 0.01
*RST: 0.75
Default unit: V

Usage: Asynchronous command

BUS<m>:DDR:STRBe:HYSTeresis <StrobeHyst>

Sets a value for the hysteresis of the strobe.

Suffix:
<m> 1..4

Parameters:
<StrobeHyst> Range: 0 to 10
Increment: 0.01
*RST: 0.35
Default unit: V

Usage: Asynchronous command

BUS<m>:DDR:STRBe:SOURce <StrobeSource>

Selects the source for the strobe signal.

Suffix:
<m> 1..4

Parameters:

<StrobeSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

*RST: C2W1

Usage: Asynchronous command

BUS<m>:DDR:STRBe:THReshold <StrobeThreshold>

Sets the threshold value for the digitization of the strobe line.

Suffix:

<m> 1..4

Parameters:

<StrobeThreshold> Range: -10 to 10
Increment: 0.01
*RST: 0
Default unit: V

Usage: Asynchronous command

BUS<m>:DDR:BITRate <Bitrate>

Declares the nominal bit rate of the data signal for the DDR eye diagram analysis.

Suffix:

<m> 1..4

Parameters:

<Bitrate> Range: 1000 to 10E+9
Increment: 100E+3
*RST: 800E+6
Default unit: bps

Usage: Asynchronous command

24.17.28.2 Decode results

In all `BUS<m>:DDR:RESult:FRAMe<n>` commands, the suffix <m> selects the serial bus and the suffix <n> selects the frame number in the decode table.

BUS<m>:DDR:RESult:FCOunt? 2390
BUS<m>:DDR:RESult:FRAMe<n>:FLD<o>:FVAL? 2390
BUS<m>:DDR:RESult:FRAMe<n>:FLD<o>:STARt? 2390
BUS<m>:DDR:RESult:FRAMe<n>:FLD<o>:STATus? 2391
BUS<m>:DDR:RESult:FRAMe<n>:FLD<o>:STOP? 2391
BUS<m>:DDR:RESult:FRAMe<n>:FLD<o>:VAL? 2391
BUS<m>:DDR:RESult:FRAMe<n>:NWRDs? 2392
BUS<m>:DDR:RESult:FRAMe<n>:STARt? 2392

BUS<m>:DDR:RESult:FRAMe<n>:STATe?.....2392
BUS<m>:DDR:RESult:FRAMe<n>:STOP?.....2393
BUS<m>:DDR:RESult:FRAMe<n>:TYPE?.....2393

BUS<m>:DDR:RESult:FCOunt?

Returns the number of decoded frames in the selected serial bus.

Suffix:
<m> 1..4

Return values:
<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:DDR:RESult:FRAMe<n>:FLD<o>:FVAL?

Returns the formatted value of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:
<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:
<FrameByteValue>

Usage: Query only

BUS<m>:DDR:RESult:FRAMe<n>:FLD<o>:STARt?

Returns the start time of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:
<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:DDR:RESult:FRAMe<n>:FLD<o>:STATus?

Returns the status of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<State> OK | UNKNown | INComplete | LENGth
*RST: OK

Usage: Query only

BUS<m>:DDR:RESult:FRAMe<n>:FLD<o>:STOP?

Returns the stop time of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:DDR:RESult:FRAMe<n>:FLD<o>:VAL?

Returns the value of the field with the selected field number within the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *
Selects the frame number.

<o> *
Selects the field number.

Return values:

<FrameByteValue> Range: 0 to 0
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:DDR:RESult:FRAMe<n>:NWRDs?

Returns the number of words for the selected serial bus and frame number.

Suffix:

<m> 1..4

<n> *

Return values:

<NumWords> Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only

BUS<m>:DDR:RESult:FRAMe<n>:STARt?

Returns the start time of the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

Return values:

<Start> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:DDR:RESult:FRAMe<n>:STATe?

Returns the status of the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

Return values:

<State> OK | UNKNown | INComplete | LENGth

*RST: OK

Usage: Query only

BUS<m>:DDR:RESult:FRAMe<n>:STOP?

Returns the stop time of the frame with the selected frame number (for the selected serial bus).

Suffix:

<m> 1..4

<n> *

Return values:

<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

BUS<m>:DDR:RESult:FRAMe<n>:TYPE?

Returns the frame type for the selected serial bus and frame number.

Suffix:

<m> 1..4

<n> *

Usage: Query only

24.17.28.3 Search settings

There are two commands for each parameter, that you can use for defining the DDR search settings.

For example, to set the *Frame type =Read > Field =Data >Data* value you can use one of the following commands:

- `SEARch:TRIGger:DDR:FRAMe1:ENABle:DMIN 01100`
  Defines the parameter by using the index <m> for the frame number and <n> for the field number.
- `SEARch:TRIGger:DDR:DMIN "Read", "Data", 01100`
  Defines the parameter by using the frame and field name.

SEARch:TRIGger:DDR:ERENable....2394
SEARch:TRIGger:DDR:ERRor<m>:ENABle....2394
SEARch:TRIGger:DDR:FIENable....2394
SEARch:TRIGger:DDR:FRENable....2395
SEARch:TRIGger:DDR:DMAX....2395
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DMAX....2395
SEARch:TRIGger:DDR:DMIN....2395
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DMIN....2395
SEARch:TRIGger:DDR:DOPerator....2396
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DOPerator....2396
SEARch:TRIGger:DDR:FRENable....2396
SEARch:TRIGger:DDR:FRAMe<m>:ENABle....2396
SEARch:TRIGger:DDR:IMAX....2397
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IMAX....2397
SEARch:TRIGger:DDR:IMIN....2397
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IMIN....2397
SEARch:TRIGger:DDR:IOPerator....2397
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IOPerator....2397

SEARch:TRIGger:DDR:ERENable <SearchName>,<ErrorName>, <Enabler>
SEARch:TRIGger:DDR:ERRor<m>:ENABle <SearchName>,<Enable>
SEARch:TRIGger:DDR:ERRor<m>:ENABle? <SearchName>

Defines the error type to be searched for. You can search for all error types in parallel.

Suffix:
<m> *
Specifies the error.

Parameters:
<Enable> ON | OFF
*RST: ON

Parameters for setting and query:
<SearchName> The following errors are available:
`Frame Incomplete`
`Length Error`

Example: `SEARch:TRIGger:DDR:ERRor1:ENABle ON`
Enables searching for Frame Incomplete error.
`SEARch:TRIGger:DDR:ERENable "Length Error",ON`
Enables searching for Length Error.

SEARch:TRIGger:DDR:FIENable <SearchName>,<Frame>, <Field>, <Enabler>

Setting parameters:
<SearchName>

<Frame>

<Field>

<Enabler> ON | OFF

Usage: Setting only

SEARch:TRIGger:DDR:FRENable <SearchName>,<Frame>, <Enabler>

Setting parameters:
<SearchName>

<Frame>

<Enabler> ON | OFF

Usage: Setting only

SEARch:TRIGger:DDR:DMAX <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DMAX <SearchName>,<DataMax>
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DMAX? <SearchName>

Sets the end value of a data pattern range if `SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DOPerator` is set to `INRange` or `OORange`.

Suffix:
<m> *

<n> *

Parameters:
<DataMax>

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DDR:DMIN <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DMIN <SearchName>,<DataMin>
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DMIN? <SearchName>

Specifies the data pattern to be searched, or sets the start value of a data pattern range to be searched.

Suffix:
<m> *

<n> *

Parameters:
<DataMin> Specifies the name of the generic trigger setting frame.

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DDR:DOPerator <SearchName>,<Frame>, <Field>, <Operator>
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DOPerator
<SearchName>,<DataOperator>
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DOPerator? <SearchName>

Sets the operator to set a specific data pattern to be searched in the selected field of the selected frame.

Suffix:
<m> *

<n> *

Parameters:
<DataOperator> EQUal | NEQual | LTHan | LETHan | GTHan | GETHan | INRange | OORange

EQUal | NEQual | LTHan | LETHan | GTHan | GETHan
Equal, Not equal, Less than, Less than or equal, Greater than, Greater than or equal. These conditions require one endpoint value to be set using `SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DMIN`.

INRange | OORange
In range, out of range. These conditions require a range of endpoint values to be set using `SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DMIN` and `SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DMAX`.

*RST: EQUal

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DDR:FRENable <SearchName>,<Frame>, <Enabler>
SEARch:TRIGger:DDR:FRAMe<m>:ENABle <SearchName>,<Enable>
SEARch:TRIGger:DDR:FRAMe<m>:ENABle? <SearchName>

Enables or disables the specific frame to be searched for.

Suffix:
<m> *

Parameters:
<Enable> ON | OFF

*RST: OFF

Parameters for setting and query:
<SearchName> The following frames are available:
`Read`
`Write`
`Read NC`
`Write NC`

SEARch:TRIGger:DDR:IMAX <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IMAX <SearchName>,<IndexMax>
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IMAX? <SearchName>

Sets the end value of the bit index range (data position range), if `SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IOPerator` is set to `INRange` or `RANGe`.

Suffix:
<m> *

<n> *

Parameters:
<IndexMax> Range: 1 to 65534
Increment: 1
*RST: 65534

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DDR:IMIN <SearchName>,<Frame>, <Field>, <Data>
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IMIN <SearchName>,<IndexMin>
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IMIN? <SearchName>

Sets the index at which the data is to be searched, or sets the start value of an index range in which the data is to be searched.

Suffix:
<m> *

<n> *

Parameters:
<IndexMin> Specifies the index value or the start value of an index range within the field.
The index range, increment and *RST values depend on the field type.

Range: 1 to 65534
Increment: 1
*RST: 1

Parameters for setting and query:
<SearchName>

SEARch:TRIGger:DDR:IOPerator <SearchName>,<Frame>, <Field>, <Operator>
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IOPerator
<SearchName>,<IndexOperator>
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IOPerator? <SearchName>

Sets the operator for the index for searching in the selected field of the selected frame.

Suffix:
<m> *

<n> *

Parameters:

<IndexOperator> EQUal | INRange | RANGe

EQUal
This condition requires one endpoint value to be set using `SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IMIN`.

INRange | RANGe
This condition requires a range of endpoint values to be set using `SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IMIN` and `SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IMAX`.

*RST: INRange

Parameters for setting and query:
<SearchName>

24.17.28.4 Search results

In all `SEARch:RESult:DDR:FRAMe<m>` commands, the suffix <m> selects the frame number in the list of search results.

SEARch:RESult:DDR:FCOunt?....2398
SEARch:RESult:DDR:FRAMe<m>:FLD<n>:FVAL?....2398
SEARch:RESult:DDR:FRAMe<m>:FLD<n>:STARt?....2399
SEARch:RESult:DDR:FRAMe<m>:FLD<n>:STATus?....2399
SEARch:RESult:DDR:FRAMe<m>:FLD<n>:STOP?....2400
SEARch:RESult:DDR:FRAMe<m>:FLD<n>:VAL?....2400
SEARch:RESult:DDR:FRAMe<m>:NWRDs?....2400
SEARch:RESult:DDR:FRAMe<m>:STARt?....2401
SEARch:RESult:DDR:FRAMe<m>:STATe?....2401
SEARch:RESult:DDR:FRAMe<m>:STOP?....2401
SEARch:RESult:DDR:FRAMe<m>:TYPE?....2402

SEARch:RESult:DDR:FCOunt? <SearchName>

Returns the number of decoded frames in the search result.

Query parameters:
<SearchName>

Return values:

<FrameCount> Range: 0 to 100000
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:DDR:FRAMe<m>:FLD<n>:FVAL? <SearchName>

Returns the formatted value of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *
Selects the field number.

Query parameters:
<SearchName>

Return values:
<BteFormattedVal>

Usage: Query only

SEARch:RESult:DDR:FRAMe<m>:FLD<n>:STARt? <SearchName>

Returns the start time of the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *
Selects the field number.

Query parameters:
<SearchName>

Return values:
<Start>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:DDR:FRAMe<m>:FLD<n>:STATus? <SearchName>

Returns the field status of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *

Query parameters:
<SearchName>

Return values:
<State> OK | UNKNown | INComplete | LENGth
*RST: OK

Usage: Query only

SEARch:RESult:DDR:FRAMe<m>:FLD<n>:STOP? <SearchName>

Returns the stop time of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *
Selects the field number.

Query parameters:
<SearchName>

Return values:
<Stop> Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

SEARch:RESult:DDR:FRAMe<m>:FLD<n>:VAL? <SearchName>

Returns the value of the field with the selected field number within the frame with the selected frame number within the search result.

Suffix:
<m> *

<n> *
Selects the field number.

Query parameters:
<SearchName>

Return values:
<ByteValue> Range: 0 to 0
Increment: 1
*RST: 0

Usage: Query only

SEARch:RESult:DDR:FRAMe<m>:NWRDs? <SearchName>

Returns the number of words of the selected frame in the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<NumWords>	Range:	0 to 4294967295
	Increment:	1
	*RST:	0

Usage: Query only

SEARch:RESult:DDR:FRAMe<m>:STARt? <SearchName>

Returns the start time of the frame with the selected frame number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Start>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

SEARch:RESult:DDR:FRAMe<m>:STATe? <SearchName>

Returns the status of the frame with the selected frame number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<State>	OK | UNKNown | INComplete | LENGth	
	*RST:	OK

Usage: Query only

SEARch:RESult:DDR:FRAMe<m>:STOP? <SearchName>

Returns the stop time of the frame with the selected frame number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Return values:

<Stop>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0
	Default unit:	s

Usage: Query only

SEARch:RESult:DDR:FRAMe<m>:TYPE? <SearchName>

Returns the frame type for the selected frame number within the search result.

Suffix:
<m> *

Query parameters:
<SearchName>

Usage: Query only

24.17.28.5 Eye diagram analysis

The suffix EYE<m> selects the eye diagram to which the command applies. You can configure and analyze up to four eye diagrams in parallel.

There are two ways to generate eye diagrams on the R&S RTO6:

- Standard eye diagrams, which are a superposition of repetitively sampled digital data.
- Advanced eye diagrams, which are generated by superimposing waveform slices from a multi-bit acquisition. To use advanced eye diagrams, option R&S RTO6-K91 is required.

Basic setup

EYE<m>[:STATe] 2403
EYE<m>:SOURce 2403
EYE<m>:HORizontal[:UNIT] 2403
EYE<m>:HORizontal:ABSolute:RANGe 2403
EYE<m>:HORizontal:ABSolute:POSition 2404
EYE<m>:HORizontal:UINTerval:RANGe 2404
EYE<m>:HORizontal:UINTerval:POSition 2404
EYE<m>:TIMReference:CLOCk:EDGE 2405
EYE<m>:TIMReference:CLOCk:FACTor 2405
EYE<m>:TIMReference:CLOCk:FREQuency 2405
EYE<m>:TIMReference:CLOCk:FREStimation 2406
EYE<m>:TIMReference:CLOCk:LEVel 2406
EYE<m>:TIMReference:CLOCk:OFEStimation 2406
EYE<m>:TIMReference:CLOCk:OFFSet 2406
EYE<m>:TIMReference:CLOCk:SAMTime 2407
EYE<m>:TIMReference:CLOCk:SOURce 2407

EYE<m>:TIMReference:PROTocol:SELect 2407
EYE<m>:TIMReference[:SOURce] 2407
EYE<m>:MEASurement:MAXPrecision 2408

EYE<m>[:STATe] <State>

Activates or deactivates the eye diagram.

Suffix:
<m> 1..4

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

EYE<m>:SOURce <MainSource>

Selects the waveform from which the eye diagram is generated (data source). For analog channels, peak detect, envelope, and average waveforms are not supported.

Suffix:
<m> 1..4

Parameters:
<MainSource> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

Usage: Asynchronous command

EYE<m>:HORizontal[:UNIT] <TimeUnits>

Sets the method to define the horizontal scale of the eye diagram.

Suffix:
<m> 1..4

Parameters:
<TimeUnits> ATIMe | UINTerval

ATIMe
Absolute time

UINTerval
Unit intervals. You can use unit interval, if the nominal frequency of the clock signal is known.

*RST: UINTerval

Usage: Asynchronous command

EYE<m>:HORizontal:ABSolute:RANGe <HorizontalRange>

Sets the time range that is covered by the eye diagram.

Suffix:
<m> 1..4

Parameters:
<HorizontalRange> Range: 100E-12 to 100
Increment: 1E-9
*RST: 200E-9
Default unit: s

Usage: Asynchronous command

EYE<m>:HORizontal:ABSolute:POSition <HorizPosition>

Sets the place of the zero point in the diagram, in seconds. The zero point is the alignment point on which the slice timestamps are superimposed.

Suffix:
<m> 1..4

Parameters:
<HorizPosition> Range: -100 to 100
Increment: 1E-9
*RST: 0
Default unit: s

Usage: Asynchronous command

EYE<m>:HORizontal:UINTerval:RANGe <RangeUI>

Sets the number of unit intervals that are visible in the eye diagram.

Suffix:
<m> 1..4

Parameters:
<RangeUI> Range: 1E-6 to 100E+9
Increment: 0.01
*RST: 2

Usage: Asynchronous command

EYE<m>:HORizontal:UINTerval:POSition <PositionUI>

Sets the zero point in the diagram, in unit intervals. The zero point is the alignment point on which the segment timestamps are superimposed.

Suffix:
<m> 1..4

Parameters:

<PositionUI>
Range: -100E+9 to 100E+9
Increment: 0.01
*RST: 0.5
Default unit: UI

Usage: Asynchronous command

EYE<m>:TIMReference:CLOCk:EDGE <EdgeType>

The clock edge sets the point in time when the state of the data signal is checked.

Suffix:

<m> 1..4

Parameters:

<EdgeType> POSitive | NEGative | EITHer

*RST: EITHer

Usage: Asynchronous command

EYE<m>:TIMReference:CLOCk:FACTor <Factor>

Sets a clock multiplier factor.

Available if `EYE<m>:TIMReference[:SOURce]` is set to `CLOCk`.

Suffix:

<m> 1..4

Parameters:

<Factor>
Range: 1 to 100
Increment: 1
*RST: 1

Usage: Asynchronous command

EYE<m>:TIMReference:CLOCk:FREQuency <Frequency>

Sets the clock frequency for the explicit clock signal.

Available if `EYE<m>:TIMReference[:SOURce]` is set to `CLOCk`.

Suffix:

<m> 1..4

Parameters:

<Frequency>
Range: 1000 to 10E+9
Increment: 10
*RST: 1E+9
Default unit: Hz

Usage: Asynchronous command

EYE<m>:TIMReference:CLOCk:FREStimation

Executes an estimation of the signal for the clock frequency.

Available if `EYE<m>:TIMReference[:SOURce]` is set to `CLOCk`.

Suffix:
<m> 1..4

Usage: Event

EYE<m>:TIMReference:CLOCk:LEVel <Level>

Sets the vertical reference level of the timing reference source.

Available if `EYE<m>:TIMReference[:SOURce]` is set to `CLOCk`.

Suffix:
<m> 1..4

Parameters:
<Level> UPPer | MIDDle | LOWer
*RST: MIDDle

Usage: Asynchronous command

EYE<m>:TIMReference:CLOCk:OFEStimation

Executes an estimation of the signal for the clock offset.

Available if `EYE<m>:TIMReference[:SOURce]` is set to `CLOCk`.

Suffix:
<m> 1..4

Usage: Event

EYE<m>:TIMReference:CLOCk:OFFSet <Offset>

Sets a clock offset for the explicit clock signal.

Available if `EYE<m>:TIMReference[:SOURce]` is set to `CLOCk`.

Suffix:
<m> 1..4

Parameters:
<Offset> Range: -1000 to 1000
Increment: 0.01
*RST: 0
Default unit: UI

Usage: Asynchronous command

EYE<m>:TIMReference:CLOCk:SAMTime <SamplingTime>

Sets a sampling time for the explicit clock signal, an offset for the clock edge in relation to the bit start. The clock edge sets the beginning of the unit interval. The sampling time is a number between 0 and 1. Value 0 sets the clock edge to the beginning of the bit period; value 0.5 sets the clock edge to the middle of the bit period.

Available if `EYE<m>:TIMReference[:SOURce]` is set to `CLOCk`.

Suffix:
<m> 1..4

Parameters:
<SamplingTime> Range: 0 to 1
Increment: 0.01
*RST: 0.5
Default unit: UI

Usage: Asynchronous command

EYE<m>:TIMReference:CLOCk:SOURce <SourceType>

Selects the source of the clock channel, if `EYE<m>:TIMReference[:SOURce]` is set to `CLOCk`.

Suffix:
<m> 1..4

Parameters:
<SourceType> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

Usage: Asynchronous command

EYE<m>:TIMReference:PROTocol:SELect <SerialBusIndex>

Selects the active bus for the eye analysis, if `EYE<m>:TIMReference[:SOURce]` is set to `PROTocol`.

Suffix:
<m> 1..4

Parameters:
<SerialBusIndex> SBUS1 | SBUS2 | SBUS3 | SBUS4
*RST: SBUS1

Usage: Asynchronous command

EYE<m>:TIMReference[:SOURce] <ReferenceSource>

Selects the source of the reference signal used for the eye diagram.

Suffix:
<m> 1..4

Parameters:
<ReferenceSource> CLOCk | CDR | PROTocol
*RST: CLOCk

Usage: Asynchronous command

EYE<m>:MEASurement:MAXPrecision <EnabClassifier>

If this function is enabled, the rising and falling transitions for the eye measurmens "Eye rise time", "Eye fall time" and "Duty cycle distortion" are separated.

Suffix:
<m> 1..4

Parameters:
<EnabClassifier> ON | OFF
*RST: OFF

Usage: Asynchronous command

Display

EYE<m>:MSLices <MaxSliceCount>

Sets the number of waveform slices for a single acquisition.

Suffix:
<m> 1..4

Parameters:
<MaxSliceCount> Range: 1 to 1000000
Increment: 1
*RST: 2000

Usage: Asynchronous command

EYE<m>:DISPlay:MODE <EyestripeMode>

Defines when the eye stripe is shown: always, never, or during mask tests.

Suffix:
<m> 1..4

Parameters:
<EyestripeMode> ALWays | MTONly | NEVer
MTONly: Mask test only
*RST: MTONly

Usage: Asynchronous command

EYE<m>:DISPlay:PERSistence:INFinite <State>

If persistence is enabled (`EYE<m>:DISPlay:PERSistence:STATe`), each new data point in the diagram area remains on the screen infinitely until this command is set to `OFF`.

Suffix:
<m> 1..4

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

EYE<m>:DISPlay:PERSistence:RESet

Resets the display, removing persistent values.

Suffix:
<m> 1..4

Usage: Event
Asynchronous command

EYE<m>:DISPlay:PERSistence:STATe <State>

If enabled, each new data point in the diagram area remains on the screen for the duration defined using `EYE<m>:DISPlay:PERSistence:TIME`, or as long as `EYE<m>:DISPlay:PERSistence:INFinite` is enabled.

If disabled, the signal value is only displayed as long as it actually occurs.

Suffix:
<m> 1..4

Parameters:
<State> ON | OFF
*RST: ON

Usage: Asynchronous command

EYE<m>:DISPlay:PERSistence:TIME <Time>

If persistence is enabled (`EYE<m>:DISPlay:PERSistence:STATe`), each new data point in the diagram area remains on the screen for the duration defined here.

Suffix:
<m> 1..4

Parameters:

<Time>	Range:	0.05 to 50
	Increment:	0.05
	*RST:	0.05
	Default unit:	s

Usage: Asynchronous command

Qualify: gate settings

EYE<m>:QUALify:GATE[:STATe]....2410
EYE<m>:QUALify:GATE:COUPling....2410
EYE<m>:QUALify:GATE:MODE....2410
EYE<m>:QUALify:GATE:ABSolute:STARt....2411
EYE<m>:QUALify:GATE:ABSolute:STOP....2411
EYE<m>:QUALify:GATE:RELative:STARt....2411
EYE<m>:QUALify:GATE:RELative:STOP....2411
EYE<m>:QUALify:GATE:CURSor....2412
EYE<m>:QUALify:GATE:ZDIagram....2412

EYE<m>:QUALify:GATE[:STATe] <State>

Enables the gate.

Suffix:
<m> 1..4

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

EYE<m>:QUALify:GATE:COUPling <CouplingMode>

Sets the gate coupling. To use cursor lines or zoom areas as gate limits, the cursor or zoom must be defined before.

Suffix:
<m> 1..4

Parameters:
<CouplingMode> NONE | CURSor | ZOOM
*RST: NONE

Usage: Asynchronous command

EYE<m>:QUALify:GATE:MODE <Mode>

Defines whether the gate limit is defined in absolute or relative values when no gate coupling is set.

Suffix:
<m> 1..4

Parameters:
<Mode> ABS | REL
*RST: ABS

Usage: Asynchronous command

EYE<m>:QUALify:GATE:ABSolute:STARt <Start>
EYE<m>:QUALify:GATE:ABSolute:STOP <Stop>

Set the absolute time values for gate start and gate end.

Suffix:
<m> 1..4

Parameters:
<Start>, <Stop> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

Example: Gate definition for eye diagram 2 in absolute time values:

```
EYE2:QUALify:GATE:COUPling NONE
EYE2:QUALify:GATE:MODE ABS
EYE2:QUALify:GATE:ABSolute:STARt -0.0002
EYE2:QUALify:GATE:ABSolute:STOP 0.0002
```

Usage: Asynchronous command

EYE<m>:QUALify:GATE:RELative:STARt <RelativeStart>
EYE<m>:QUALify:GATE:RELative:STOP <RelativeStop>

Set the relative values for gate start and gate end.

Suffix:
<m> 1..4

Parameters:
<RelativeStart>, <RelativeStop> Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

Example: Gate definition for eye diagram 2 in relative values:

```
EYE2:QUALify:GATE:COUPling NONE
EYE2:QUALify:GATE:MODE REL
EYE2:QUALify:GATE:RELative:STARt 10
EYE2:QUALify:GATE:RElative:STOP 90
```

Usage: Asynchronous command

EYE<m>:QUALify:GATE:CURSor <Cursorset>

Selects the cursor set to be used for eye gating. The gate area is defined by the cursor lines.

Suffix:
<m> 1..4

Parameters:
<Cursorset> CURSOR1 | CURSor1 | CURSOR2 | CURSor2 | CURSOR3 | CURSor3 | CURSOR4 | CURSor4

CURSOR1 = CURSor1, CURSOR2 = CURSor2, CURSOR3 = CURSor3, CURSOR4 = CURSor4

*RST: CURSOR1

Example: Gate definition for eye diagram 2 with cursor coupling:

```
EYE2:QUALify:GATE:COUPling CURSor
EYE2:QUALify:GATE:CURSor CURSor1
```

Usage: Asynchronous command

EYE<m>:QUALify:GATE:ZDIagram <ZoomDiagram>

Selects the zoom diagram to be used for eye gating. The zoom area is is used as gate.

Suffix:
<m> 1..4

Parameters:
<ZoomDiagram> String with the name of the zoom diagram

Example: Gate definition for eye diagram 2 with zoom coupling:

```
EYE2:QUALify:GATE:COUPling ZOOM
EYE2:QUALify:GATE:ZDIagram "Zoom1"
```

Usage: Asynchronous command

Qualify: signal settings

EYE<m>:QUALify:SIGNal[:STATe] 2412
EYE<m>:QUALify:SIGNal:SOURce 2413
EYE<m>:QUALify:SIGNal:CONDition 2413
EYE<m>:QUALify:SIGNal:LSELect 2413

EYE<m>:QUALify:SIGNal[:STATe] <State>

Enables the waveform-based qualification for the eye generation.

Suffix:
<m> 1..4

Parameters:
<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

EYE<m>:QUALify:SIGNal:SOURce <Source>

Selects the waveform to be compared.

Suffix:
<m> 1..4

Parameters:
<Source> C1W1 | C2W1 | C3W1 | C4W1 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4

Usage: Asynchronous command

EYE<m>:QUALify:SIGNal:CONDition <Condition>

Selects the comparison condition for the the waveform level.

Suffix:
<m> 1..4

Parameters:
<Condition> LTHan | GTHan | INRange | OORange

*RST: GTHan

Usage: Asynchronous command

EYE<m>:QUALify:SIGNal:LSELect <LevelSelection>

Selects the threshold. You can use the lower, middle, or upper reference level. These levels are the reference levels that are defined for measurements.

Suffix:
<m> 1..4

Parameters:
<LevelSelection> UPPer | MIDDle | LOWer

*RST: MIDDle

Example: The slices of the channel 1 signal only contribute to the eye generation if the voltage level of the signal on channel 2 is higher than the upper reference level.

```
EYE:SOURce C1W1
EYE:QUALify:SIGNal:SOURce C2W1
EYE:QUALify:SIGNal:CONDition GTHan
EYE:QUALify:SIGNal:LSELect UPPer
EYE:QUALify:SIGNal ON
```

Usage: Asynchronous command

Filter

EYE<m>:FILTer:STATe 2414
EYE<m>:FILTer:BPATtern:MODE 2414
EYE<m>:FILTer:BPATtern:PREFix<1..8> 2415
EYE<m>:FILTer:BPATtern:CENTer 2415
EYE<m>:FILTer:BPATtern:SUFFix<1..5> 2415
EYE<m>:FILTer:BPATtern:SLEFt 2415
EYE<m>:FILTer:BPATtern:SRIGht 2415
EYE<m>:FILTer:BPATtern:RESet 2415
EYE<m>:FILTer:DDR:FTYPe 2415
EYE<m>:FILTer:DDR:LENerror 2416

EYE<m>:FILTer:STATe <State>

Applies the filter settings to the data signal.

Suffix:
<m> 1..4

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

EYE<m>:FILTer:BPATtern:MODE <Mode>

Selects which bits contribute to the eye generation.

Suffix:
<m> 1..4

Parameters:
<Mode> ABITs | LTRansition | CLEVel | BPATtern

ABITs
All bits contribute to the eye diagram, same as `EYE<m>:FILTer:STATe` is OFF.

LTRansition
Level transition, only bits after a signal transition contribute (0 to 1, or 1 to 0).

CLEVel
Constant level, only bits without a preceding signal transition contribute (0 to 0, or 1 to 1).

BPATtern
Only bits contribute that match the defined pattern before and after the center eye.

*RST: ABITs

Usage: Asynchronous command

EYE<m>:FILTer:BPATtern:PREFix<1..8> <Bit>
EYE<m>:FILTer:BPATtern:CENTer <BitValue>
EYE<m>:FILTer:BPATtern:SUFFix<1..5> <Bit>

Set the values of the indicated prefix, center, and suffix bits.

Suffix:
<m> 1..4

Parameters:
<Bit> ONE | ZERO | DC
DC: don't care

EYE<m>:FILTer:BPATtern:SLEFt

Shifts the pattern one bit to the left.

Suffix:
<m> 1..4

Usage: Event
Asynchronous command

EYE<m>:FILTer:BPATtern:SRIGht

Shifts the pattern one bit to the right.

Suffix:
<m> 1..4

Usage: Event
Asynchronous command

EYE<m>:FILTer:BPATtern:RESet

Resets the bit sequence filter to the default (DC for all bits).

Suffix:
<m> 1..4

Usage: Event
Asynchronous command

EYE<m>:FILTer:DDR:FTYPe <CommTyp>

Sets which frame types contribute to the eye diagram.

Suffix:
<m> 1..4

Parameters:

<CommTyp> ANY | ANYC | READ | WRITe | RDNC | WRNC

ANYC: any consecutive
RDNC: read non-consecutive
WRNC: write non-consecutive

*RST: ANY

Usage: Asynchronous command

EYE<m>:FILTer:DDR:LENerror <CommTyp>

If enabled, the length errors contribute to the eye diagram.

Suffix:

<m> 1..4

Parameters:

<CommTyp> ON | OFF

*RST: OFF

Usage: Asynchronous command

Results of eye

EYE<m>:RESult:SLICe<n>:COUNt?....2416
EYE<m>:RESult:SLICe<n>:LPOSition?....2416
EYE<m>:RESult:SLICe<n>[:POSition]?....2417
EYE<m>:RESult:TACQs?....2417
EYE<m>:RESult:TSLices?....2417

EYE<m>:RESult:SLICe<n>:COUNt?

Returns the number of filtered bits.

Suffix:

<m> 1..4

<n> *
Not relevant

Return values:

<Count>

Usage: Query only

EYE<m>:RESult:SLICe<n>:LPOSition?

Returns the position of the last slice in the acquistion.

Suffix:

<m> 1..4

<n> *

Return values:
<LastPosition>

Usage: Query only

EYE<m>:RESult:SLICe<n>[:POSition]?

Returns the position of the filtered bit in the acquisition.

Suffix:
<m> 1..4

<n> *

Return values:
<Position>
Range: -100 to 100
Increment: 1E-3
*RST: 0
Default unit: s

Usage: Query only
Asynchronous command

EYE<m>:RESult:TACQs?

Returns the total number of acquisitions accumulated into the eye.

Suffix:
<m> 1..4

Return values:
<Acquisitions>
Range: 0 to 1000000000
Increment: 1
*RST: 0

Usage: Query only
Asynchronous command

EYE<m>:RESult:TSLices?

Returns the total number of slices accumulated into the eye.

Suffix:
<m> 1..4

Return values:
<Slices>
Range: 0 to 1000000000
Increment: 1
*RST: 0

Usage: Query only
Asynchronous command

Results of mask tests on eye diagrams

MTESt:RESult:COUNt:SLICes?....2418
MTESt:RESult:COUNt:FSLices?....2418
MTESt:RESult:VIOLation<m>:COUNt?....2418
MTESt:RESult:VIOLation<m>[:SLICe]?....2419
MTESt:RESult:VIOLation<m>:STARt?....2419
MTESt:RESult:VIOLation<m>:STOP?....2419
EYE<m>:ZCOupling....2419
EYE<m>:ZDIagram....2420

MTESt:RESult:COUNt:SLICes? <MaskTestName>

Returns the number of slices that contributed to the eye diagram.

Query parameters:
<MaskTestName> String parameter

Return values:
<SlicesCompleted> Range: 0 to 100E+24
Increment: 1
*RST: 0

Usage: Query only
Asynchronous command

MTESt:RESult:COUNt:FSLices? <MaskTestName>

Returns the number of slices that violated the mask.

Query parameters:
<MaskTestName> String parameter

Return values:
<SliceHits> Range: 0 to 100E+24
Increment: 1
*RST: 0

Usage: Query only
Asynchronous command

MTESt:RESult:VIOLation<m>:COUNt? <MaskTestName>

Suffix:
<m> *
Violation index, corresponds to the number in the "Violation" column of mask test result.

Query parameters:
<MaskTestName>

Return values:
<Count>

Usage: Query only

MTESt:RESult:VIOLation<m>[:SLICe]? <MaskTestName>

Suffix:
<m> *
Violation index, corresponds to the number in the "Violation" column of mask test result.

Query parameters:
<MaskTestName>

Return values:
<Slice>
Range: 0 to 4294967295
Increment: 1
*RST: 0

Usage: Query only
Asynchronous command

MTESt:RESult:VIOLation<m>:STARt? <MaskTestName>
MTESt:RESult:VIOLation<m>:STOP? <MaskTestName>

Return the start and stop time of the slice with the indicated violation.

Suffix:
<m> *
Violation index, corresponds to the number in the "Violation" column of mask test result.

Query parameters:
<MaskTestName>

Return values:
<Time>
Range: -100 to 100
Increment: 1E-3
*RST: 0
Default unit: s

Usage: Query only
Asynchronous command

EYE<m>:ZCOupling <ZoomCoupling>

If enabled, the mask eye area is set to the limits of a zoom area.

The zoom diagram is selected using `EYE<m>:ZDIagram`.

Suffix:
<m> 1..4

Parameters:
<ZoomCoupling> ON | OFF
*RST: OFF

Usage: Asynchronous command

EYE<m>:ZDIagram <ZoomDiagram>

Selects the zoom to which the eye mask area is set if `EYE<m>:ZCOupling` is enabled.

Suffix:
<m> 1..4

Parameters:
<ZoomDiagram> String with the name of the diagram on which the zoom is based

Usage: Asynchronous command

24.18 Mixed signal option (MSO, R&S RTO6-B1)

This chaptes decribes the remote commands of MSO option R&S RTO6-B1.

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

- Digital channels....2420
- Parallel bus configuration....2423
- Digital resolution....2430
- Trigger settings for digital signals and parallel buses....2431
- MSO data....2440

24.18.1 Digital channels

All `DIGital:` commands affect only the settings of the first MSO bus (Bus1). The settings of all other parallel buses (Bus 2, 3,4) remain unchanged.

DIGital<m>:DISPlay....2421
DIGital<m>:TECHnology....2421
DIGital<m>:THReshold....2421
DIGital<m>:THCoupling....2422

DIGital<m>:HYSTeresis 2422
DIGital<m>:LABel 2423
DIGital<m>:DESKew 2423

DIGital<m>:DISPlay <Display>

Enables or disables the indicated digital channel, displays it, and enables the parallel Bus1 if the bus was disabled. That is, `BUS<m>:PARallel:DISPlay:SHDI` and `BUS<m>:PARallel:STATe` are set to ON automatically.

If another active bus already uses the selected digital channel, the instrument disables the other bus to avoid conflicts.

For Bus1, the `DIG:DISP` command has the same effect as `BUS<m>:PARallel:BIT<n>[:STATe]`. To enable digital channels for buses 2, 3 and 4, use the `BUS:PAR:BIT[:STAT]` command.

Suffix:

<m>	0..15 Number of the digital channel

Parameters:

<Display>	ON | OFF

DIGital<m>:TECHnology <Technology>

Selects the threshold voltage for various types of integrated circuits and applies it to all digital channels.

The setting affects only the settings of the first MSO bus (Bus1). You can set the technology value for all buses with `BUS<m>:PARallel:TECHnology`.

Suffix:

<m>	0..15 The suffix is irrelevant.

Parameters:

<Technology>	V15 | V25 | V165 | V125 | V09 | VM13 | V38 | V20 | V0 | MANual See `BUS<m>:PARallel:TECHnology`

DIGital<m>:THReshold <Value>

Sets the logical threshold for the channel group to which the indicated digital channel belongs. The setting affects only the settings of the first MSO bus (Bus1).

The setting affects only the settings of the first MSO bus (Bus1). You can set the threshold for all buses with `BUS<m>:PARallel:TECHnology` or `BUS<m>:PARallel:THReshold<n>`.

See also: `DIGital<m>:THCoupling`

Suffix:

<m> 0..15
Number of the digital channel.
Channel groups: 0..3; 4..7; 8..11; 12..15

Parameters:

<Value> Range: -8.0 to 8.0
Increment: 200.0e-12
*RST: 0
Default unit: V

DIGital<m>:THCoupling <State>

Sets the threshold and the hysteresis for all digital channels of parallel bus 1 to the same value.

The command `BUS<m>:PARallel:THCoupling` is used to set all buses.

Suffix:

<m> 0..15
The suffix is irrelevant.

Parameters:

<State> ON | OFF

DIGital<m>:HYSTeresis <Hysteresis>

Defines the size of the hysteresis to avoid the change of signal states due to noise for the channel group to which the indicated digital channel belongs.

The setting affects only the settings of the first MSO bus (Bus1). You can set the hysteresis for all buses with `BUS<m>:PARallel:HYSTeresis<n>`.

See also: `DIGital<m>:THCoupling`

Suffix:

<m> 0..15
Number of the digital channel
Channel groups: 0..3; 4..7; 8..11; 12..15

Parameters:

<Hysteresis> MAXIMUM | MAXimum | ROBUST | ROBust | NORMAL | NORMal

MAXIMUM = MAXimum
Maximum value that is possible and useful for the signal and its settings

ROBUST = ROBust
Different hysteresis values for falling and rising edges to avoid an undefined state of the trigger system.

NORMAL = NORMal
The instrument sets a value suitable for the signal and its settings.

DIGital<m>:LABel <Label>

Sets a name for the indicated digital channel. The name is displayed in the diagram.

The setting affects only the settings of the first MSO bus (Bus1). You can set the label for all buses with `BUS<m>:PARallel:BIT<n>:LABel`.

Suffix:
<m> 0..15
Number of the digital channel

Parameters:
<Label> String containing the channel name

DIGital<m>:DESKew <Deskew>

Sets an individual delay for each digital channel to time-align it with other digital channels. The deskew value compensates delays that are known from the circuit specifics or caused by the different length of cables.

The setting affects only the settings of the first MSO bus (Bus1). You can set the deskew for all buses with `BUS<m>:PARallel:BIT<n>:DESKew`.

Suffix:
<m> 0..15
Number of the digital channel

Parameters:
<Deskew> Range: -200.0E-09 to 200.0E-09
Increment: 200.0E-12
*RST: 0
Default unit: s

24.18.2 Parallel bus configuration

The following commands configure the four parallel buses of R&S RTO6-B1.

BUS<m>:PARallel:STATe....2424
BUS<m>:PARallel:BIT<n>[:STATe]....2424
BUS<m>:PARallel:THReshold<n>....2424
BUS<m>:PARallel:TECHnology....2425
BUS<m>:PARallel:THCoupling....2425
BUS<m>:PARallel:HYSTeresis<n>....2426
BUS<m>:PARallel:BIT<n>:DESKew....2426
BUS<m>:PARallel:DESoffset....2427
BUS<m>:PARallel:BIT<n>:LABel....2427
BUS<m>:PARallel:DATA:FORMat....2427
BUS<m>:PARallel:DISPlay:SHDI....2428
BUS<m>:PARallel:DISPlay:SHBU....2428
BUS<m>:PARallel:DISPlay:BTYP....2428
BUS<m>:PARallel:CLON....2429
BUS<m>:PARallel:CLOCk....2429

BUS<m>:PARallel:CLSLope....2429
BUS<m>:PARallel:CLEar....2429
BUS<m>:PARallel:DECTable<n>:SHOW....2430
BUS<m>:PARallel:DECTable<n>:COUNt?....2430
BUS<m>:PARallel:DECTable<n>:DATA?....2430

BUS<m>:PARallel:STATe <Enable>

Enables or disables the indicated parallel bus. The threshold settings of the bus take effect for all *active* parallel buses.

Dependencies: At least one digital channel must be enabled for the selected bus, otherwise the command does not work. The bus is enabled automatically if the first digital channel is enabled with `BUS<m>:PARallel:BIT<n>[:STATe]` or `DIGital<m>:DISPlay`.

Suffix:

<m> 1..4
Selects the parallel bus.

Parameters:

<Enable> ON | OFF
*RST: OFF

BUS<m>:PARallel:BIT<n>[:STATe] <Assigned>

Assigns the selected digital channel to the indicated bus, displays it, and enables the bus if the bus was disabled. That is, `BUS<m>:PARallel:DISPlay:SHDI` and `BUS<m>:PARallel:STATe` are set to ON automatically.

If another active bus already uses the selected digital channel, the instrument disables the other bus to avoid conflicts.

For parallel bus 1, the `BUS:PAR:BIT[:STATe]` command has the same effect as `DIGital<m>:DISPlay`.

Suffix:

<m> 1..4
Selects the parallel bus.

<n> 0..15
Selects the bit of the bus word. Each bit corresponds to a digital channel.

Parameters:

<Assigned> ON | OFF
*RST: OFF

BUS<m>:PARallel:THReshold<n> <Threshold>

Sets the logical threshold for the indicated channel group.

Alternatively, you can set the threshold with `BUS<m>:PARallel:TECHnology`. For the parallel bus 1, you can also use `DIGital<m>:THReshold`.

See also: `DIGital<m>:THCoupling`

Suffix:

<m> 1..4
Selects the parallel bus.

<n> 1..4
Selects the channel group:
1 = dig. channels 0..3;
2 = dig. channels 4..7
3 = dig. channels 8..11
4 = dig. channels 12..15

Parameters:

<Threshold> Range: -8.0 to 8.0
Increment: 200.0e-12
*RST: 0
Default unit: V

BUS<m>:PARallel:TECHnology <Technology>

Selects the threshold voltage for various types of integrated circuits and applies it to all digital channels.

Suffix:

<m> 1..4
Selects the parallel bus.

Parameters:

<Technology> V15 | V25 | V165 | V125 | V09 | VM13 | V38 | V20 | V0 | MANual

V15: TTL
V25: CMOS 5.0 V
V165: CMOS 3.3 V
V125: CMOS 2.5 V
V09: CMOS 1.85 V
VM13: ECL, -1.3 V
V38: PECL
V20: LVPECL
V0: Ground
MANual: Set a user-defined threshold value with `DIGital<m>:THReshold`

*RST: V165

BUS<m>:PARallel:THCoupling <LevelCoupling>

Sets the threshold for all digital channels of the selected bus to the same value. Also the hysteresis value is applied to all digital channels.

Tor parallel bus 1, the command `DIGital<m>:THCoupling` has the same effect.

Suffix:

<m> 1..4
The suffix is irrelevant.

Parameters:

<LevelCoupling> ON | OFF
*RST: ON

BUS<m>:PARallel:HYSTeresis<n> <Hysteresis>

Defines the size of the hysteresis for the channel group to avoid the change of signal states due to noise.

For the parallel bus 1, you can also use `DIGital<m>:HYSTeresis`.

Suffix:

<m> 1..4
Selects the parallel bus.

<n> 1..4
Selects the channel group:
1 = dig. channels 0..3;
2 = dig. channels 4..7
3 = dig. channels 8..11
4 = dig. channels 12..15

Parameters:

<Hysteresis> MAXIMUM | MAXimum | ROBUST | ROBust | NORMAL | NORMal

MAXIMUM = MAXimum
Maximum value that is possible and useful for the signal and its settings

ROBUST = ROBust
Different hysteresis values for falling and rising edges to avoid an undefined state of the trigger system.

NORMAL = NORMal
The instrument sets a value suitable for the signal and its settings.

BUS<m>:PARallel:BIT<n>:DESKew <Deskew>

Sets an individual delay for each digital channel to time-align it with other digital channels. The deskew value compensates delays that are known from the circuit specifics or caused by the different length of cables.

For the parallel bus 1, you can also use `DIGital<m>:DESKew`.

Suffix:

<m> 1..4
Selects the parallel bus.

<n> 0..15
Selects the bit of the bus word. Each bit corresponds to a digital channel.

Parameters:

<Deskew> Range: -200E-9 to 200E-9
Increment: 200E-12
*RST: 0
Default unit: s

BUS<m>:PARallel:DESoffset <DeskewOffset>

Sets a general delay for all digital channels. The resulting deskew of a digital channel is the sum of this general value and the individual value set with `BUS<m>:PARallel:BIT<n>:DESKew`.

Suffix:

<m> 1..4
Selects the parallel bus.

Parameters:

<DeskewOffset> Range: -200E-9 to 200E-9
Increment: 200E-12
*RST: 0
Default unit: s

BUS<m>:PARallel:BIT<n>:LABel <Label>

Sets a name for the indicated digital channel. The name is displayed in the diagram.

For the parallel bus 1, you can also use `DIGital<m>:LABel`.

Suffix:

<m> 1..4
Selects the parallel bus.

<n> 0..15
Selects the bit of the bus word. Each bit corresponds to a digital channel.

Parameters:

<Label> String containing the channel name

BUS<m>:PARallel:DATA:FORMat <DataFormat>

Sets the number format for decoded data values on the display for the indicated parallel bus.

It sets also the number representation for remote data transfer with `BUS<m>:PARallel:DATA[:VALues]?`.

Suffix:

<m> 1..4

Parameters:

<DataFormat> HEX | OCT | BIN | ASCII | ASCii | SIGN | USIG

*RST: HEX

BUS<m>:PARallel:DISPlay:SHDI <ShwDigSigns>

If enabled, the selected digital signals are shown in the diagram. Each channel is displayed as a logic signal.

See also: `DIGital<m>:DISPlay`

Suffix:

<m> 1..4
Selects the parallel bus.

Parameters:

<ShwDigSigns> ON | OFF

*RST: OFF

BUS<m>:PARallel:DISPlay:SHBU <ShowBus>

Shows or hides the indicated parallel bus. If enabled, the resulting bus signal and bus values are displayed in the diagram.

Suffix:

<m> 1..4
Selects the parallel bus.

Parameters:

<ShowBus> ON | OFF

*RST: OFF

BUS<m>:PARallel:DISPlay:BTYP <BusRepres>

Selects the display type of the indicated parallel bus.

Suffix:

<m> 1..4
Selects the parallel bus.

Parameters:

<BusRepres> COMB | ANALog

COMB
Displays the decoded bus signal with bus values.

ANALog
Displays the bus value as amplitude, similar to an analog waveform.

*RST: COMB

BUS<m>:PARallel:CLON <Clocked>

Defines if the bus is a clocked bus - one of the digital channels serves as clock of the bus.

Suffix:
<m> 1 | 2
Selects the parallel bus. The clocked bus is available on parallel buses 1 and 2.

Parameters:
<Clocked> ON | OFF
*RST: OFF

BUS<m>:PARallel:CLOCk <ClockSource>

Selects the digital channel used as clock of the indicated parallel bus.

Suffix:
<m> 1 | 2
Selects the parallel bus. The clocked bus is available on parallel buses 1 and 2.

Parameters:
<ClockSource> D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15
Clock channel
*RST: D1

BUS<m>:PARallel:CLSLope <ClockSlope>

Selects the slope of the clock signal at which all digital channels of the bus are analyzed.

Suffix:
<m> 1 | 2
Selects the parallel bus. The clocked bus is available on parallel buses 1 and 2.

Parameters:
<ClockSlope> POSitive | NEGative | EITHer
*RST: POSitive

BUS<m>:PARallel:CLEar

Removes all assigned digital channels from the bus

Suffix:
<m> 1..4
Selects the parallel bus.

Usage: Event

BUS<m>:PARallel:DECTable<n>:SHOW <ShowDecodeTable>

Available only if `BUS<m>:PARallel:CLON` is set to `ON`.

If enabled, a results box opens with decoded values of the bus signal and its time. You can query the number of values (`BUS<m>:PARallel:DECTable<n>:COUNt?`) and their data value (`BUS<m>:PARallel:DECTable<n>:DATA?`).

Suffix:
<m> 1..4

<n> 1..5

Parameters:
<ShowDecodeTable> ON | OFF
*RST: OFF

BUS<m>:PARallel:DECTable<n>:COUNt?

Returns the number of clock edges in the decode table.

Suffix:
<m> 1..4

<n> 1..5

Usage: Query only

BUS<m>:PARallel:DECTable<n>:DATA?

Returns the data of the decoded values of the bus signal and its time.

Suffix:
<m> 1..4

<n> 1..5

Usage: Query only

24.18.3 Digital resolution

ACQuire:DRESolution?....2430
ACQuire:POINts:DVALue?....2431

ACQuire:DRESolution?

Returns the current digital resolution of the digital channels.

Return values:
<DigRes> Default unit: s

Usage: Query only
Asynchronous command

ACQuire:POINts:DVALue?

Returns the current digital record length used by each digital channel.

Return values:

<DigReclgth>	Range:	1000 to 200E6
	Increment:	2
	*RST:	1000
	Default unit:	pts

Usage: Query only
Asynchronous command

24.18.4 Trigger settings for digital signals and parallel buses

In all `TRIGger<m>:PARallel` commands, the trigger suffix <m> is always 1 and can be omitted. It selects the trigger event: Only the A-trigger is available for triggering on digital signals and parallel buses.

- General commands....2431
- Edge trigger....2433
- Width trigger....2434
- Timeout trigger....2435
- Setup & Hold trigger....2436
- State trigger....2437
- Pattern trigger....2437
- Serial pattern trigger....2440

24.18.4.1 General commands

TRIGger<m>:SOURce[:SELect]....2431
TRIGger<m>:PARallel:TYPE....2432
TRIGger<m>:PARallel:SETHold:CSOurce[:VALue]....2433
TRIGger<m>:PARallel:STATe:CSOurce:VALue....2433
TRIGger<m>:PARallel:SPATtern:CSOurce[:VALue]....2433
TRIGger<m>:PARallel:EDGE:EXPRession[:DEFine]....2433
TRIGger<m>:PARallel:WIDTh:EXPRession[:DEFine]....2433
TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine]....2433
TRIGger<m>:PARallel:STATe:EXPRession[:DEFine]....2433
TRIGger<m>:PARallel:PATTern:EXPRession[:DEFine]....2433
TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine]....2433

TRIGger<m>:SOURce[:SELect] <SourceDetailed>

Selects the source of the trigger signal.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger
Available values depend on the selected trigger source. For input channels CHAN1...4, a trigger sequence can be configured.
For all other trigger sources, only suffix 1 is allowed.

Parameters:

<SourceDetailed> CHAN1 | CHANnel1 | CHAN2 | CHANnel2 | CHAN3 | CHANnel3 | CHAN4 | CHANnel4 | EXTernanalog | SBUS | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | LOGIC | MSOB1 | MSOB2 | MSOB3 | MSOB4 | Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4
Input channels

EXTernanalog
External analog signal connected to the External Trigger Input.
For this source, only the analog edge trigger is available.

SBUS
Serial bus

D0...D15
Digital channels (option R&S RTO6-B1)
See also: Chapter 24.18.4, "Trigger settings for digital signals and parallel buses", on page 2431

LOGIc
Logic combination of digital channels, used as trigger source (option R&S RTO6-B1)

MSOB1 | MSOB2 | MSOB3 | MSOB4
Parallel bus (option R&S RTO6-B1)

Z1V1 | Z1V2 | Z1V3 | Z1V4 | Z1I1 | Z1I2 | Z1I3 | Z1I4 | Z2V1 | Z2V2 | Z2V3 | Z2V4 | Z2I1 | Z2I2 | Z2I3 | Z2I4
Input channels of R&S RT-ZVCmulti-channel power probe. Only available in the A-trigger with trigger type EDGE.

DIFF1 | DIFF2 | COMMON1 | COMMON2
Differential signals

*RST: CHAN1

Usage: Asynchronous command

TRIGger<m>:PARallel:TYPE <Type>

Selects the trigger type to trigger on digital channels and parallel buses.

To trigger on analog channels or the external trigger input, use `TRIGger<m>:TYPE`.

Parameters:

<Type> EDGE | WIDTh | TIMeout | SETHold | STATe | PATTern | SERPattern |

SERPattern = serial pattern trigger

*RST: EDGE

Usage: Asynchronous command

TRIGger<m>:PARallel:SETHold:CSOurce[:VALue] <ClockSource>
TRIGger<m>:PARallel:STATe:CSOurce:VALue <ClockSource>
TRIGger<m>:PARallel:SPATtern:CSOurce[:VALue] <ClockSource>

Selects the digital channel of the clock signal.

Parameters:

<ClockSource> D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15

*RST: D0

Usage: Asynchronous command

TRIGger<m>:PARallel:EDGE:EXPRession[:DEFine] <LogicalExpr>
TRIGger<m>:PARallel:WIDTh:EXPRession[:DEFine] <LogicalExpr>
TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine] <LogicalExpr>
TRIGger<m>:PARallel:STATe:EXPRession[:DEFine] <LogicalExpr>
TRIGger<m>:PARallel:PATTern:EXPRession[:DEFine] <LogicalExpr>
TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine] <LogicalExpr>

Defines a logical combination of several digital channels as trigger condition if `TRIGger<m>:SOURce[:SELect]` is set to LOGIC.

Parameters:

<LogicalExpr> String with logical expression

Example: `TRIGger:PARallel:EDGE:EXPRession 'D1 and D2'`

Usage: Asynchronous command

24.18.4.2 Edge trigger

See also:

- `TRIGger<m>:PARallel:EDGE:EXPRession[:DEFine]` on page 2433

TRIGger<m>:PARallel:EDGE:SLOPe.....2433

TRIGger<m>:PARallel:EDGE:SLOPe <Slope>

Defines the edge - the state transition - of the signal to trigger on a single digital channel (a logic bit), or a logical combination of digital channels.

Parameters:
<Slope> POSitive | NEGative | EITHer
*RST: POSitive

Usage: Asynchronous command

24.18.4.3 Width trigger

See also:

- `TRIGger<m>:PARallel:WIDTh:EXPRession[:DEFine]` on page 2433

TRIGger<m>:PARallel:WIDTh:POLarity....2434
TRIGger<m>:PARallel:WIDTh:RANGe....2434
TRIGger<m>:PARallel:WIDTh:WIDTh....2434
TRIGger<m>:PARallel:WIDTh:DELTa....2435

TRIGger<m>:PARallel:WIDTh:POLarity <Polarity>

Sets the polarity of a pulse. When triggering on a positive pulse, the trigger event occurs on the high to low transition of the pulse if the timing condition is true. When triggering on a negative pulse, the trigger event occurs on the low to high transition of the pulse if the timing condition is true.

Parameters:
<Polarity> POSitive | NEGative | EITHer
*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:PARallel:WIDTh:RANGe <RangeMode>

Selects how the range of a pulse width is defined:

Parameters:
<RangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin | OUTSide
Pulses inside or outside a given time range. The time range *Width ± Delta* is defined by `TRIGger<m>:PARallel:WIDTh:WIDTh` and `TRIGger<m>:PARallel:WIDTh:DELTa`.

SHORter | LONGer
Pulses shorter or longer than a given width defined by `TRIGger<m>:PARallel:WIDTh:WIDTh`

*RST: WITHin

Usage: Asynchronous command

TRIGger<m>:PARallel:WIDTh:WIDTh <Width>

Sets the limit for the pulse width.

The effect depends on `TRIGger<m>:PARallel:WIDTh:RANGe`.

- For the ranges SHORter and LONGer, the width defines the maximum and minimum pulse width, respectively.
- For the ranges WITHin and OUTSide, the width defines the center of a range which is defined by the limits `TRIGger<m>:PARallel:WIDTh:DELTa`.

Parameters:

<Width>
Range: 200E-12 to 10000
Increment: 200E-9
*RST: 5E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:PARallel:WIDTh:DELTa <WidthDelta>

Defines a range around the given width value. the setting is relevant if `TRIGger<m>:PARallel:WIDTh:RANGe` is set to WITHin or OUTSide. The width is set with `TRIGger<m>:PARallel:WIDTh:WIDTh`.

Parameters:

<WidthDelta>
Range: 0 to 432
Increment: 600E-12
*RST: 0
Default unit: s

Usage: Asynchronous command

24.18.4.4 Timeout trigger

See also:

- `TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine]` on page 2433

TRIGger<m>:PARallel:TIMeout:RANGe 2435
TRIGger<m>:PARallel:TIMeout:TIME 2436

TRIGger<m>:PARallel:TIMeout:RANGe <TimeoutMode>

Sets the state condition.

Parameters:

<TimeoutMode> HIGH | LOW | EITHer

HIGH
The level of a digital channel stays above the threshold, or the logical expression for LOGic trigger source is true.

LOW
The level of a digital channel stays below the threshold, or the logical expression for LOGic trigger source is false.

EITHer
The signal state remains unchanged.

*RST: HIGH

Usage: Asynchronous command

TRIGger<m>:PARallel:TIMeout:TIME <Time>

Defines the time limit for the timeout at which the instrument triggers.

Parameters:

<Time>
Range: 100E-12 to 10000
Increment: 100E-9
*RST: 100E-9
Default unit: s

Usage: Asynchronous command

24.18.4.5 Setup & Hold trigger

See also:

- `TRIGger<m>:PARallel:SETHold:CSOurce[:VALue]` on page 2433

TRIGger<m>:PARallel:SETHold:CSOurce:EDGE....2436
TRIGger<m>:PARallel:SETHold:HTIMe....2436
TRIGger<m>:PARallel:SETHold:STIMe....2437

TRIGger<m>:PARallel:SETHold:CSOurce:EDGE <ClockEdge>

Sets the edge of the clock signal. The crossing of the clock edge and the logical threshold defines the time reference point for the setup and hold time measurement.

Parameters:

<ClockEdge>
POSitive | NEGative | EITHer
*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:PARallel:SETHold:HTIMe <HoldTime>

Sets the minimum time *after* the clock edge while data should be stable and not change its state.

See also: "Hold time" on page 1003

Parameters:

<HoldTime>
Range: -99.8E-9 to 100E-9
Increment: 1E-9
*RST: 0
Default unit: s

Usage: Asynchronous command

TRIGger<m>:PARallel:SETHold:STIMe <SetupTime>

Sets the minimum time *before* the clock edge while data should be stable and not change its state.

See also: "Setup time" on page 1003

Parameters:

<SetupTime>
Range: -99.8E-9 to 100E-9
Increment: 1E-9
*RST: 0
Default unit: s

Usage: Asynchronous command

24.18.4.6 State trigger

See also:

- `TRIGger<m>:PARallel:STATe:CSOurce:VALue` on page 2433
- `TRIGger<m>:PARallel:STATe:EXPRession[:DEFine]` on page 2433

TRIGger<m>:PARallel:STATe:CSOurce:EDGE.....2437
TRIGger<m>:PARallel:STATe:BIT<0..15>.....2437

TRIGger<m>:PARallel:STATe:CSOurce:EDGE <Slope>

Sets the edge of the clock signal. The crossing of the clock edge and the logical threshold defines the time at which the logical states and the bus value are analyzed.

Parameters:

<Slope>
POSitive | NEGative | EITHer
*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:PARallel:STATe:BIT<0..15> <Bit>

Sets the required state for each digital channel that is used in the bus.

Parameters:

<Bit>
HIGH | LOW | DONTCARE | DONTcare
Bit value: 1 (HIGH), 0 (LOW), or X (DONTCARE = DONTcare)

Usage: Asynchronous command

24.18.4.7 Pattern trigger

TRIGger<m>:PARallel:PATTern:BIT<0..15> <Bit>

Sets the required state for each digital channel that is used in the bus.

Parameters:

<Bit> HIGH | LOW | DONTCARE | DONTcare

Bit value: 1 (HIGH), 0 (LOW), or X (DONTCARE = DONTcare)

Usage: Asynchronous command

TRIGger<m>:PARallel:PATTern:MODE <Mode>

Sets the mode of the timing condition.

Parameters:

<Mode> OFF | TIMeout | WIDTh

OFF
No timing condition, only the logical pattern condition is relevant.

TIMeout
Defines a minimum time qualification to avoid triggering on unstable or transitional conditions. Use `TRIGger<m>:PARallel:PATTern:TIMeout:MODE` and `TRIGger<m>:PARallel:PATTern:TIMeout[:TIME]` to specify the timeout.

WIDTh
Sets a pulse width as timing condition. The pulse starts when the pattern comes true, and the trigger event occurs when the pattern comes false during the specified time limit. Use `TRIGger<m>:PARallel:PATTern:WIDTh:RANGe`, `TRIGger<m>:PARallel:PATTern:WIDTh[:WIDTh]`, and `TRIGger<m>:PARallel:PATTern:WIDTh:DELTa` to specify the width.

*RST: OFF

Usage: Asynchronous command

TRIGger<m>:PARallel:PATTern:TIMeout:MODE <TimeoutMode>

Sets the state condition for the timeout qualification if `TRIGger<m>:PARallel:PATTern:MODE` is set to TIMeout. To set the time limit, use `TRIGger<m>:PARallel:PATTern:TIMeout[:TIME]`.

Parameters:

<TimeoutMode> HIGH | LOW | EITHer

HIGH: The pattern stays true for the specified time.
LOW: The pattern stays false for the specified time.
EITHER: The pattern remains unchanged for the specified time.

*RST: HIGH

Usage: Asynchronous command

TRIGger<m>:PARallel:PATTern:TIMeout[:TIME] <Time>

Defines the time limit for the timeout at which the instrument triggers.

Parameters:

<Time>

Range: 100E-12 to 10000
Increment: 100E-9
*RST: 100E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:PARallel:PATTern:WIDTh:RANGe <WidthRangeMode>

Selects how the range of a pulse width is defined if `TRIGger<m>:PARallel:PATTern:MODE` is set to WIDTh.

Parameters:

<WidthRangeMode> WITHin | OUTSide | SHORter | LONGer

WITHin
Triggers when the pattern comes false inside a given time range. The time limit is defined by `TRIGger<m>:PARallel:PATTern:WIDTh[:WIDTh]` and `TRIGger<m>:PARallel:PATTern:WIDTh:DELTa` (*Width ± Delta*).

OUTSide
Triggers when the pattern comes false before or after the given time range. The time limit definition is the same as for WITHin range.

SHORter | LONGer
Triggers when the pattern comes false before or after the given width has expired. Width is set with `TRIGger<m>:PARallel:PATTern:WIDTh[:WIDTh]`.

*RST: WITHin

Usage: Asynchronous command

TRIGger<m>:PARallel:PATTern:WIDTh[:WIDTh] <Width>

The effect depend on the setting of the `TRIGger<m>:PARallel:PATTern:WIDTh:RANGe`command.

For the ranges SHOrter and LONGer, the width defines the maximum and minimum time limit, respectively.

For the ranges WITHin and OUTSide, the width defines the center of a range which is defined by the limits "±Delta".

Parameters:

<Width>

Range: 100E-12 to 10000
Increment: 100E-9
*RST: 5E-9
Default unit: s

Usage: Asynchronous command

TRIGger<m>:PARallel:PATTern:WIDTh:DELTa <WidthDelta>

Defines a range around the width value set with `TRIGger<m>:PARallel:PATTern:WIDTh[:WIDTh]`.

Parameters:

<WidthDelta>
Range: 0 to 432
Increment: 500E-12
*RST: 0
Default unit: s

Usage: Asynchronous command

24.18.4.8 Serial pattern trigger

See also:

- `TRIGger<m>:PARallel:SPATtern:CSOurce[:VALue]` on page 2433
- `TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine]` on page 2433

TRIGger<m>:PARallel:SPATtern:CSOurce:EDGE <ClockEdge>

Sets the edge of the clock signal. The bit value is determined at the crossing of the clock edge and the logical threshold.

Parameters:

<ClockEdge>
POSitive | NEGative | EITHer
*RST: POSitive

Usage: Asynchronous command

TRIGger<m>:PARallel:SPATtern:PATTern <Pattern>

Defines the serial bit string on which to trigger.

Parameters:

<Pattern>
Numeric or string pattern, see Chapter 24.4.6, "Bit pattern parameter", on page 1286. The bit value X (don't care) is not allowed.

Usage: Asynchronous command

24.18.5 MSO data

To export data of digital channels and parallel buses to file, use the following commands:

- `EXPort:WAVeform:SOURce` on page 1649
- `EXPort:WAVeform:NAME` on page 1651
- `EXPort:WAVeform:SAVE` on page 1651

The remote data transfer from the instrument to the controlling computer is performed using the following commands:

DIGital<m>:DATA:HEADer? 2441
DIGital<m>:DATA[:VALues]? 2441
BUS<m>:PARallel:DATA:FORMat 2441
BUS<m>:PARallel:DATA:HEADer? 2442
BUS<m>:PARallel:DATA[:VALues]? 2442

DIGital<m>:DATA:HEADer?

Returns the header of digital channel data

Table 24-16: Header data

Position	Meaning	Example
1	XStart, acquisition time before trigger, in s	-5E-008 = - 50 ns
2	XStop, acquisition time after trigger, in s	5E-008 = 50 ns
3	Record length of the waveform in Samples	1000
4	Number of values per sample interval. For digital data the result is 1.	1

Suffix:
<m> 0..15
Number of the digital channel

Usage: Query only

DIGital<m>:DATA[:VALues]?

Returns the data of the indicated digital channel for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

Suffix:
<m> 0..15
Selects the digital channel.

Return values:
<Data> List of data (0 and 1)

Usage: Query only

BUS<m>:PARallel:DATA:FORMat <DataFormat>

Sets the number format for decoded data values on the display for the indicated parallel bus.

It sets also the number representation for remote data transfer with `BUS<m>:PARallel:DATA[:VALues]?`.

Suffix:
<m> 1..4

Parameters:

<DataFormat> HEX | OCT | BIN | ASCII | ASCii | SIGN | USIG

*RST: HEX

BUS<m>:PARallel:DATA:HEADer?

Returns the header data of the indicated bus.

For a detailed description, see `DIGital<m>:DATA:HEADer?`.

Suffix:

<m> 1..4
Selects the parallel bus.

Usage: Query only

Firmware/software: V 2.40

BUS<m>:PARallel:DATA[:VALues]?

Returns the data of the indicated parallel bus.

Requirements:

- `BUS<m>:PARallel:STATe` is set to `ON`.
- `BUS<m>:PARallel:DISPlay:SHBU` is set to `ON`.
- Number format is set with `BUS<m>:PARallel:DATA:FORMat`.

Suffix:

<m> 1..4
Selects the parallel bus.

Return values:

<Data> List of values according to the format setting.

Example:

```
BUS:PAR:STAT ON
BUS:PAR:DISP:SHBU ON
BUS:PAR:DISP:BTYP COMB
BUS:PAR:DATA:FORMat HEX
BUS:PAR:DATA:VAL?
```

Usage: Query only

24.19 Waveform generator (option R&S RTO6-B6)

24.19.1 Waveform generator setup

24.19.1.1 General

WGENerator<m>:ACOPy....2443
WGENerator<m>[:ENABle]....2443
WGENerator<m>:SOURce....2443
WGENerator<m>:PRESet....2443

WGENerator<m>:ACOPy

Copies all settings from Gen1/Gen2 and applies them to Gen2/Gen1.

Suffix:
<m> 1..2

Usage: Event

WGENerator<m>[:ENABle] <State>

Enables the waveform generator and outputs the waveform.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF

WGENerator<m>:SOURce <OperationMode>

Sets the operation mode for the the waveform generator.

Suffix:
<m> 1..2

Parameters:
<OperationMode> FUNCgen | MODulation | SWEep | ARBGenerator
*RST: FUNCgen

WGENerator<m>:PRESet

Sets the parameters of the waveform generator to their default values.

Suffix:
<m> 1..2

Usage: Event

24.19.1.2 Function generator

WGENerator<m>:FUNCtion[:SELect] 2444
WGENerator<m>:FREQuency 2444
WGENerator<m>:PERiod 2444
WGENerator<m>:FUNCtion:PULSe[:WIDTh] 2445
WGENerator<m>:FUNCtion:RAMP[:SYMMetry] 2445
WGENerator<m>:FUNCtion:SQUare:DCYCle 2445

WGENerator<m>:FUNCtion[:SELect] <FunctionType>

Sets the type of waveform to be generated for the function generator.

Suffix:
<m> 1..2

Parameters:
<FunctionType> SINusoid | SQUare | RAMP | DC | PULSe | SINC | CARDiac | GAUSs | LORNtz | EXPRise | EXPFall

SINC: Cardial sine

*RST: SINusoid

WGENerator<m>:FREQuency <Frequency>

Sets the frequency of the waveform. The available frequency range depends on the selected function type.

Suffix:
<m> 1..2

Parameters:
<Frequency> Range: 1E-3 to 100E+6
Increment: 1
*RST: 1E+6
Default unit: Hz

WGENerator<m>:PERiod <Period>

Sets the period of the waveform. The available period range depends on the selected function type.

Suffix:
<m> 1..2

Parameters:
<Period> Range: 8E-9 to 1000
Increment: 1
*RST: 1E-6
Default unit: s

WGENerator<m>:FUNCtion:PULSe[:WIDTh] <PulseWidth>

Sets the pulse duration for a pulse waveform.

Suffix:
<m> 1..2

Parameters:
<PulseWidth> Range: 16.5E-9 to 1000, actual range depends on the selected frequency
Increment: 1
*RST: 500E-9
Default unit: s

WGENerator<m>:FUNCtion:RAMP[:SYMMetry] <RampSymmetry>

Sets the symmetry for a ramp waveform.

Suffix:
<m> 1..2

Parameters:
<RampSymmetry> Range: 0 to 100
Increment: 1
*RST: 50
Default unit: %

WGENerator<m>:FUNCtion:SQUare:DCYCle <SquareDutyCycle>

Sets the duty cycle for the generated square waveform.

Suffix:
<m> 1..2

Parameters:
<SquareDutyCycle> Range: 0.01 to 99.99 at minimum frequency, actual range depends on the selected frequency
Increment: 1
*RST: 50
Default unit: %

24.19.1.3 Modulation

WGENerator<m>:MODulation:TYPE....2446
WGENerator<m>:MODulation:AM:DCYCle....2446
WGENerator<m>:MODulation:AM:DEPTh....2446
WGENerator<m>:MODulation:AM:FREQuency....2447
WGENerator<m>:MODulation:AM:SYMMetry....2447
WGENerator<m>:MODulation:CARRier:FREQuency....2447
WGENerator<m>:MODulation:CARRier:PERiod....2447
WGENerator<m>:MODulation:AM[:FUNCtion]....2448

WGENerator<m>:MODulation:FM:DCYCle....2448
WGENerator<m>:MODulation:FM:DEViation....2448
WGENerator<m>:MODulation:FM:FREQuency....2448
WGENerator<m>:MODulation:FM:SYMMetry....2449
WGENerator<m>:MODulation:FM[:FUNCtion]....2449
WGENerator<m>:MODulation:FSK:FONE....2449
WGENerator<m>:MODulation:FSK:FTWO....2450
WGENerator<m>:MODulation:FSK[:RATE]....2450
WGENerator<m>:MODulation:PWM:DCYCle....2450
WGENerator<m>:MODulation:PWM:DEPTh....2450
WGENerator<m>:MODulation:PWM:FREQuency....2451
WGENerator<m>:MODulation:PWM:SYMMetry....2451
WGENerator<m>:MODulation:PWM[:FUNCtion]....2451

WGENerator<m>:MODulation:TYPE <ModulationType>

Sets the modulation type.

Suffix:
<m> 1..2

Parameters:
<ModulationType> AM | FM | PWM | FSK
*RST: AM

WGENerator<m>:MODulation:AM:DCYCle <SquareDutyCycle>

Sets the duty cycle of a square waveform for amplitude modulation, if `WGENerator<m>:MODulation:TYPE` is set to `AM`.

Suffix:
<m> 1..2

Parameters:
<SquareDutyCycle> Range: 10 to 90
Increment: 1
*RST: 50
Default unit: %

WGENerator<m>:MODulation:AM:DEPTh <Depth>

Sets the amplitude modulation depth.

Suffix:
<m> 1..2

Parameters:
<Depth> Range: 0 to 100
Increment: 1
*RST: 50
Default unit: %

WGENerator<m>:MODulation:AM:FREQuency <Frequency>

Sets the frequency of the modulation waveform for amplitude modulation, if `WGENerator<m>:MODulation:TYPE` is set to `AM`.

Suffix:
<m> 1..2

Parameters:
<Frequency>
Range: 1E-3 to 1E+6
Increment: 1
*RST: 1000
Default unit: Hz

WGENerator<m>:MODulation:AM:SYMMetry <RampSymmetry>

Sets the symmetry, the percentage of time the ramp modulation waveform is rising, for amplitude modulation, if `WGENerator<m>:MODulation:TYPE` is set to `AM`.

Suffix:
<m> 1..2

Parameters:
<RampSymmetry>
Range: 0 to 100
Increment: 1
*RST: 50
Default unit: %

WGENerator<m>:MODulation:CARRier:FREQuency <CarrierFreq>

Sets the frequency of the carrier signal for a modulation waveform.

Suffix:
<m> 1..2

Parameters:
<CarrierFreq>
Range: 1E-3 to 100E+6
Increment: 10
*RST: 1E+6
Default unit: Hz

WGENerator<m>:MODulation:CARRier:PERiod <CarrierPeriod>

Sets the period of the carrier signal for a modulation waveform.

Suffix:
<m> 1..2

Parameters:

<CarrierPeriod>	Range:	8E-9 to 1000
	Increment:	10
	*RST:	1E-6
	Default unit:	s

WGENerator<m>:MODulation:AM[:FUNCtion] <SignalType>

Sets the type of the modulation signal for the amplitude modulation, if `WGENerator<m>:MODulation:TYPE` is set to `AM`.

Suffix:

<m>	1..2

Parameters:

<SignalType>	SINusoid | SQUare | RAMP	
	*RST:	SINusoid

WGENerator<m>:MODulation:FM:DCYCle <SquareDutyCycle>

Sets the duty cycle of a square waveform for frequency modulation, if `WGENerator<m>:MODulation:TYPE` is set to `FM`.

Suffix:

<m>	1..2

Parameters:

<SquareDutyCycle>	Range:	10 to 90
	Increment:	1
	*RST:	50
	Default unit:	%

WGENerator<m>:MODulation:FM:DEViation <Deviation>

Sets the frequency modulation deviation.

Suffix:

<m>	1..2

Parameters:

<Deviation>	Range:	1E-3 to 10E+6
	Increment:	1
	*RST:	1000
	Default unit:	Hz

WGENerator<m>:MODulation:FM:FREQuency <Frequency>

Sets the frequency of the modulation waveform for frequency modulation, if `WGENerator<m>:MODulation:TYPE` is set to `FM`.

Suffix:
<m> 1..2

Parameters:
<Frequency> Range: 1E-3 to 1E+6
Increment: 1
*RST: 1000
Default unit: Hz

WGENerator<m>:MODulation:FM:SYMMetry <RampSymmetry>

Sets the symmetry, the percentage of time the ramp modulation waveform is rising, for frequency modulation, if `WGENerator<m>:MODulation:TYPE` is set to `FM`.

Suffix:
<m> 1..2

Parameters:
<RampSymmetry> Range: 0 to 100
Increment: 1
*RST: 50
Default unit: %

WGENerator<m>:MODulation:FM[:FUNCtion] <SignalType>

Sets the type of the modulation signal for the frequency modulation, if `WGENerator<m>:MODulation:TYPE` is set to `FM`.

Suffix:
<m> 1..2

Parameters:
<SignalType> SINusoid | SQUare | RAMP
*RST: SINusoid

WGENerator<m>:MODulation:FSK:FONE <Frequency1>

Sets the frequency of the carrier waveform, if `WGENerator<m>:MODulation:TYPE` is set to `FSK`.

Suffix:
<m> 1..2

Parameters:
<Frequency1> Range: 1E-3 to 100E+6
Increment: 1
*RST: 1E+6
Default unit: Hz

WGENerator<m>:MODulation:FSK:FTWO <Frequency2>

Sets the frequency of the modulated waveform, if `WGENerator<m>:MODulation:TYPE` is set to `FSK`.

Suffix:
<m> 1..2

Parameters:
<Frequency2>
Range: 1E-3 to 100E+6
Increment: 1
*RST: 1000
Default unit: Hz

WGENerator<m>:MODulation:FSK[:RATE] <Rate>

Sets the hop rate, the time before a switch from the carrier frequency set with `WGENerator<m>:MODulation:FSK:FONE` and the modulation frequency set with `WGENerator<m>:MODulation:FSK:FTWO`.

Suffix:
<m> 1..2

Parameters:
<Rate>
Range: 1E-3 to 1E+6
Increment: 1
*RST: 1000
Default unit: Hz

WGENerator<m>:MODulation:PWM:DCYCle <SquareDutyCycle>

Sets the duty cycle of a square waveform for pulse width modulation, if `WGENerator<m>:MODulation:TYPE` is set to `PWM`.

Suffix:
<m> 1..2

Parameters:
<SquareDutyCycle>
Range: 10 to 90
Increment: 1
*RST: 50
Default unit: %

WGENerator<m>:MODulation:PWM:DEPTh <Depth>

Sets the modulation depth for pulse width modulation.

Suffix:
<m> 1..2

Parameters:

<Depth>	Range:	0 to 100
	Increment:	1
	*RST:	50
	Default unit:	%

WGENerator<m>:MODulation:PWM:FREQuency <Frequency>

Sets the frequency of the modulation waveform for pulse width modulation, if `WGENerator<m>:MODulation:TYPE` is set to `PWM`.

Suffix:

<m> 1..2

Parameters:

<Frequency>	Range:	1E-3 to 1E+6
	Increment:	1
	*RST:	1000
	Default unit:	Hz

WGENerator<m>:MODulation:PWM:SYMMetry <RampSymmetry>

Sets the symmetry, the percentage of time the ramp modulation waveform is rising, for pulse width modulation, if `WGENerator<m>:MODulation:TYPE` is set to `PWM`.

Suffix:

<m> 1..2

Parameters:

<RampSymmetry>	Range:	0 to 100
	Increment:	1
	*RST:	50
	Default unit:	%

WGENerator<m>:MODulation:PWM[:FUNCtion] <SignalType>

Sets the type of the modulation signal for the pulse width modulation, if `WGENerator<m>:MODulation:TYPE` is set to `PWM`.

Suffix:

<m> 1..2

Parameters:

<SignalType> SINusoid | SQUare | RAMP

*RST: SINusoid

24.19.1.4 Sweep

WGENerator<m>:SWEep:FSTart....2452
WGENerator<m>:SWEep:TIME....2452
WGENerator<m>:SWEep[:FEND]....2452

WGENerator<m>:SWEep:FSTart <StartFrequency>

Sets the start frequency of the sweep range.

Suffix:
<m> 1..2

Parameters:
<StartFrequency> Range: 1E-3 to 100E+6
Increment: 1
*RST: 1000
Default unit: Hz

WGENerator<m>:SWEep:TIME <Time>

Sets the duration of the sweep.

Suffix:
<m> 1..2

Parameters:
<Time> Range: 1E-3 to 500
Increment: 1
*RST: 1E-3
Default unit: s

WGENerator<m>:SWEep[:FEND] <StopFrequency>

Sets the stop frequency of the sweep range.

Suffix:
<m> 1..2

Parameters:
<StopFrequency> Range: 1E-3 to 100E+6
Increment: 1
*RST: 1E+6
Default unit: Hz

24.19.1.5 ARB

WGENerator<m>:ARBGen:COPY 2453
WGENerator<m>:ARBGen:MULTichannel:IMPort 2453
WGENerator<m>:ARBGen:MULTichannel:NAME 2453
WGENerator<m>:ARBGen:MULTichannel:OPEN 2453
WGENerator<m>:ARBGen:NAME 2454
WGENerator<m>:ARBGen:OPEN 2454
WGENerator<m>:ARBGen:RUNMode 2454
WGENerator<m>:ARBGen:SAMPles? 2454
WGENerator<m>:ARBGen:SELect 2455

WGENerator<m>:ARBGen:SRATe....2455
WGENerator<m>:ARBGen[:SOURce]....2455
WGENerator<m>:ARBGen:RUNSingle....2456

WGENerator<m>:ARBGen:COPY

Loads the waveform from the selected signal source (`WGENerator<m>:ARBGen:SELect`).

Suffix:
<m> 1..2

Usage: Event

WGENerator<m>:ARBGen:MULTichannel:IMPort <SavedWfmSrc>

Assigns a waveform from the multichannel file to the arbitrary waveform generator.

Suffix:
<m> 1..2

Parameters:
<SavedWfmSrc> WFM1 | WFM2 | WFM3 | WFM4
*RST: WFM1

Example:
```
WGENerator1:ARBGen:MULTichannel:NAME
ArbMultichannelCurve_2017-02-16_01.bin
WGENerator1:ARBGen:MULTichannel:IMPort WF1
WGENerator1:ARBGen:MULTichannel:OPEN
```

WGENerator<m>:ARBGen:MULTichannel:NAME <FilePath>

Defines the path and the multichannel arbitrary waveform file to be imported. If not path is given, the default path `C:\Users\Public\Public Documents\Rohde-Schwarz\RTx\RefWaveforms` is used.

Suffix:
<m> 1..2

Parameters:
<FilePath>

Example: See `WGENerator<m>:ARBGen:MULTichannel:IMPort` on page 2453.

WGENerator<m>:ARBGen:MULTichannel:OPEN

Loads the arbitrary waveform.

Suffix:
<m> 1..2

Example: See `WGENerator<m>:ARBGen:MULTichannel:IMPort` on page 2453.

Usage: Event

WGENerator<m>:ARBGen:NAME <FilePath>

Sets the file path and the file for an arbitrary waveform, if `WGENerator<m>:ARBGen[:SOURce]` on page 2455 is set to `ARBitrary`.

If not path is given, the default path `C:\Users\Public\Public Documents\Rohde-Schwarz\RTx\RefWaveforms` is used.

This command is only relevant for files with one waveform defined. For multichannel arbitrary waveform files, see `WGENerator<m>:ARBGen:MULTichannel:OPEN`.

Suffix:
<m> 1..2

Parameters:
<FilePath>

WGENerator<m>:ARBGen:OPEN

Loads the arbitrary waveform, that is selected with the `WGENerator<m>:ARBGen:NAME` command.

This command is only relevant for files with one waveform defined. For multichannel arbitrary waveform files, see `WGENerator<m>:ARBGen:MULTichannel:OPEN`.

Suffix:
<m> 1..2

Usage: Event

WGENerator<m>:ARBGen:RUNMode <RunMode>

Sets the duration for which the signal of the arbitrary generator will be output after the trigger event.

Suffix:
<m> 1..2

Parameters:
<RunMode> SINGle | CONTinuous
*RST: CONTinuous

WGENerator<m>:ARBGen:SAMPles?

Displays the number of samples for the arbitrary waveform.

Suffix:
<m> 1..2

Return values:
<NumSamples> Range: 0 to 128000000
Increment: 10
*RST: 0
Default unit: pts

Usage: Query only

WGENerator<m>:ARBGen:SELect <ScopeSignSrc>

Selects the oscilloscope source, from which the arbitrary signal is loaded, if `WGENerator<m>:ARBGen[:SOURce]` is set to `SCOPe`.

Suffix:
<m> 1..2

Parameters:
<ScopeSignSrc> NONE | C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | R1 | R2 | R3 | R4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

*RST: NONE

WGENerator<m>:ARBGen:SRATe <SampleRate>

Sets the sample rate for the arbitrary waveform.

Suffix:
<m> 1..2

Parameters:
<SampleRate> Range: 1 to 250E+6
Increment: 10
*RST: 1E+6
Default unit: Sa/s

WGENerator<m>:ARBGen[:SOURce] <WaveformSource>

Sets the arbitrary waveform source.

Suffix:
<m> 1..2

Parameters:
<WaveformSource> ARBitrary | SCOPe

ARBitrary
Enables you to load an existing arbitrary file.

SCOPe
Enables you to load a scope waveform.

*RST: ARBitrary

WGENerator<m>:ARBGen:RUNSingle

Executes a single period of the arbitrary signal generator, for `WGENerator<m>:ARBGen:RUNMode SINGle`.

Suffix:
<m> 1..2

Usage: Event

24.19.2 Pattern generator setup

PGENerator:BITRate....2456
PGENerator:ENABle....2456
PGENerator:FILE:OPEN....2456
PGENerator:FILE[:NAME]....2456
PGENerator:HLEVel....2457
PGENerator:PRESet....2457
PGENerator:RUNMode....2457
PGENerator:RUNSingle....2457

PGENerator:BITRate <BitRate>

Sets the number of transmitted bits per second for the pattern generator.

Parameters:
<BitRate> Range: 1 to 40E+6
Increment: 10
*RST: 1E+6
Default unit: bps

PGENerator:ENABle <State>

Enables the pattern generator and outputs the waveform.

Parameters:
<State> ON | OFF

PGENerator:FILE:OPEN

Loads the specified pattern file to the instrument.

Usage: Event

PGENerator:FILE[:NAME] <FilePath>

Specifies a file path to open a pattern generator file.

Parameters:
<FilePath> String with the name of the file.

PGENerator:HLEVel <HighLevel>

Sets the high level of the pattern generator signal.

Parameters:
<HighLevel> Range: 1.2 to 5
Increment: 0.1
*RST: 1.2
Default unit: V

PGENerator:PRESet

Sets the default pattern generator settings.

Usage: Event

PGENerator:RUNMode <RunMode>

Sets the duration for which the signal of the pattern generator will be output after the trigger event.

Parameters:
<RunMode> SINGle | CONTinuous
*RST: CONTinuous

PGENerator:RUNSingle

Executes a single period of the pattern generator, for `PGENerator:RUNMode SINGle`.

Usage: Event

24.19.3 Output setup

WGENerator<m>:OUTPut[:LOAD] 2457
WGENerator<m>:VOLTage[:VPP] 2458
WGENerator<m>:VOLTage:DCLevel 2458
WGENerator<m>:VOLTage:HIGH 2458
WGENerator<m>:VOLTage:INVersion 2459
WGENerator<m>:VOLTage:LOW 2459
WGENerator<m>:VOLTage:OFFSet 2459

WGENerator<m>:OUTPut[:LOAD] <Load>

Sets the load of the DUT at its connection.

Suffix:
<m> 1..2

Parameters:
<Load> FIFTy | HIZ
HIZ: high input impedance
*RST: HIZ

WGENerator<m>:VOLTage[:VPP] <Amplitude>

Sets the amplitude, peak to peak voltage, of the output waveform. This is defined as the voltage difference between the maximum (`WGENerator<m>:VOLTage:HIGH`) and the minimum (`WGENerator<m>:VOLTage:LOW`) voltage levels.

The value is set for the currently selected load (`WGENerator<m>:OUTPut[:LOAD]`). If the load is changed, the value of the amplitude is adapted to this new setting.

Suffix:
<m> 1..2

Parameters:
<Amplitude> Range: 0.01 to 12
Increment: 0.1
*RST: 1
Default unit: V

WGENerator<m>:VOLTage:DCLevel <DCLevel>

Sets the DC level of the generated DC signal.

Suffix:
<m> 1..2

Parameters:
<DCLevel> Range: -5.995 to 5.995
Increment: 0.1
*RST: 0
Default unit: V

WGENerator<m>:VOLTage:HIGH <High>

Sets the high signal level of the output waveform.

Suffix:
<m> 1..2

Parameters:
<High> Range: -5.99 to 6
Increment: 0.1
*RST: 0.5
Default unit: V

WGENerator<m>:VOLTage:INVersion <Inversion>

Enables inversion, which swaps the two poles of the waveform.

Suffix:
<m> 1..2

Parameters:
<Inversion> ON | OFF
*RST: OFF

WGENerator<m>:VOLTage:LOW <Low>

Sets the low signal level of the output waveform.

Suffix:
<m> 1..2

Parameters:
<Low> Range: -6 to 5.99
Increment: 0.1
*RST: -0.5
Default unit: V

WGENerator<m>:VOLTage:OFFSet <Offset>

Sets a voltage offset.

Suffix:
<m> 1..2

Parameters:
<Offset> Range: -5.995 to 5.995
Increment: 0.1
*RST: 0
Default unit: V

24.19.4 Noise setup

WGENerator<m>:MODulation:NDCLevel....2459
WGENerator<m>:MODulation:NLABsolute?....2460
WGENerator<m>:MODulation:NLPCent....2460
WGENerator<m>:MODulation:NOISe....2460

WGENerator<m>:MODulation:NDCLevel <LevelDC>

Sets the DC noise level.

Suffix:
<m> 1..2

Parameters:

<LevelDC>	Range:	0 to 12
	Increment:	0.1
	*RST:	0.01
	Default unit:	V

WGENerator<m>:MODulation:NLABsolute?

Queries the level of the noise in volts.

Suffix:

<m>	1..2

Return values:

<LevelAbs>	Range:	0 to 12
	Increment:	0.1
	*RST:	0
	Default unit:	V

Usage: Query only

WGENerator<m>:MODulation:NLPCent <LevelPct>

Sets the noise level in percentage of the amplitude.

Suffix:

<m>	1..2

Parameters:

<LevelPct>	Range:	0 to 100
	Increment:	1
	*RST:	0
	Default unit:	%

WGENerator<m>:MODulation:NOISe <State>

Enables the adding of noise to the waveform.

Suffix:

<m>	1..2

Parameters:

<State>	ON | OFF	
	*RST:	OFF

24.19.5 DC offset alignment

GENerator:ALIGnment:DC:ABORt....2461
GENerator:ALIGnment:DC:RESult:DATE?....2461
GENerator:ALIGnment:DC:RESult:TIME?....2461
GENerator:ALIGnment:DC:RESult[:STATe]?....2461
GENerator:ALIGnment:DC[:STARt]....2461

GENerator:ALIGnment:DC:ABORt

Aborts a DC offset alignment.

Usage: Event

GENerator:ALIGnment:DC:RESult:DATE?

Queries the date of the last performed DC offset alignment.

Return values:
<Date>

Usage: Query only

GENerator:ALIGnment:DC:RESult:TIME?

Queries the time of the last performed DC offset alignment.

Return values:
<Time>

Usage: Query only

GENerator:ALIGnment:DC:RESult[:STATe]?

Queries the result of a DC offset alignment.

Return values:
<State> PASSed | FAILed | NOALigndata
*RST: NOALigndata

Usage: Query only

GENerator:ALIGnment:DC[:STARt]

Starts a DC offset alignment.

Usage: Event

24.19.6 Coupling & sync

WGENerator<m>:COUPling:STATe....2462
WGENerator<m>:COUPling:AMPLitude....2462
WGENerator<m>:COUPling:PHASeshift....2462
WGENerator<m>:COUPling[:FREQuency]....2463
WGENerator<m>:COUPling:NOISe....2463
WGENerator<m>:COUPling:AWForm....2463
GENerator:SYNC[:COMBination]....2463

WGENerator<m>:COUPling:STATe <CplEnabSt>

Enables the coupling of the generators, with the selected set of parameters: amplitude, frequency and noise.

Suffix:
<m> 1..2

Parameters:
<CplEnabSt> ON | OFF
*RST: OFF

WGENerator<m>:COUPling:AMPLitude <CoupleAmplitude>

Enables the coupling of all amplitude parameters of the generators.

Suffix:
<m> 1..2
Specifies the master generator.

Parameters:
<CoupleAmplitude> ON | OFF
*RST: OFF

WGENerator<m>:COUPling:PHASeshift <PhaseShift>

Sets the phase shift between the waveform of Gen1 and Gen2 when the frequency parameters of the two waveforms are coupled.

Suffix:
<m> 1..2
Specifies the master generator.

Parameters:
<PhaseShift> Range: -180 to 180
Increment: 1
*RST: 0
Default unit: °

WGENerator<m>:COUPling[:FREQuency] <CoupleFrequency>

Enables the coupling of all frequency parameters of the generators.

Suffix:
<m> 1..2
Specifies the master generator.

Parameters:
<CoupleFrequency> ON | OFF
*RST: OFF

WGENerator<m>:COUPling:NOISe <CoupleNoise>

Couples all noise parameters of the generators.

Suffix:
<m> 1..2

Parameters:
<CoupleNoise> ON | OFF
*RST: OFF

WGENerator<m>:COUPling:AWForm <CplArbitraryWfm>

Enables the coupling of the arbitrary waveform of the generators.

Suffix:
<m> 1..2

Parameters:
<CplArbitraryWfm> ON | OFF
*RST: OFF

GENerator:SYNC[:COMBination] <Combination>

Sets which signals generated from the waveform generator are synchronized.

Parameters:
<Combination> NONE | GEN12 | G1PG | G2PG | G12PG
GEN12: Gen1 and Gen 2
GEN1PG/GEN2PG: Gen1/Gen 2 and Patt Gen
GEN12PG: Gen1,Gen 2 and Patt Gen
*RST: NONE

24.20 Pulse Source and TDR/TDT Analysis (options R&S RTO6-B7/K130)

24.20.1 Pulse source (option R&S RTO6-B7)

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

PSRC[:STAT]....2464
PSRC:OUTPutlow....2464
PSRC:REPRate....2465
PSRC:DUTYcycle....2465
PSRC:CLOCkmode....2465
PSRC:RST....2465

PSRC[:STAT] <State>

Switches the pulse output on and off.

Parameters:

<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

PSRC:OUTPutlow <OutputLowLevel>

Sets the low level of the output pulse.

Parameters:

<OutputLowLevel> Range: -0.2 to -0.05
Increment: 0.01
*RST: -0.05
Default unit: V

Usage: Asynchronous command

PSRC:REPRate <RepetitionRate>

Sets the pulse frequency, the repetition rate of the pulse.

Parameters:

<RepetitionRate> Available values depend on the selected clock mode:
Locked: 5/10/20/50/100/200/500 Hz, 1/5/10/25/50/100/250 MHz
Free running: 5/10/20/50/100/200/500 Hz, 1/5/10/25/50 MHz

*RST: 1E+6
Default unit: Hz

Usage: Asynchronous command

PSRC:DUTYcycle <DutyCycle>

Sets the duty cycle of the pulse, which is the ratio of the positive pulse width to the period of the signal.

Parameters:

<DutyCycle> For repetition rates > 5 MHz, the value is fixed at 50 %.

Range: 10 to 90
Increment: 10
*RST: 50
Default unit: %

Usage: Asynchronous command

PSRC:CLOCkmode <ClockMode>

Sets the dependency of the pulse clock on the instrument's reference clock.

Parameters:

<ClockMode> LOCKed | FREerunning

LOCKed
The pulse source is locked to the reference clock of the instrument.

FREerunning
The clock of the pulse source is independent. Deviations of the system do not affect the pulse clock, and deterministic conditions are avoided.

*RST: FREerunning

Usage: Asynchronous command

PSRC:RST

Resets the pulse source to the default state.

Usage: Event
Asynchronous command

24.20.2 TDR/TDT analysis

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

24.20.2.1 Calibration

TDRT:DEFault 2466
TDRT:CALibration:OPEN:NAME 2466
TDRT:CALibration:OPEN[:EXECute] 2466
TDRT:TYPE 2467
TDRT:PULSe:OUT 2467
TDRT:PULSe:REFerence 2467
TDRT:SLENgth 2467
TDRT:CALibration:SAVE:NAME 2468
TDRT:CALibration:SAVE[:EXECute] 2468

TDRT:DEFault

Starts the measurement with default correction values instead of calibrating the setup. Default values are rough values without time reference.

Usage: Event
Asynchronous command

TDRT:CALibration:OPEN:NAME <Path>

Sets the file name, file format and path of the TDR/TDT calibration file to be opened. The file format is always `.xml`.

Parameters:
<Path> String with path and file name with extension `.xml`

Usage: Asynchronous command

TDRT:CALibration:OPEN[:EXECute]

Opens and loads the TDR/TDT calibration from the file specified with `TDRT:CALibration:OPEN:NAME`.

Usage: Event
Asynchronous command

TDRT:TYPE <Type>

Selects the analysis type: TDR (time domain reflectometry), TDT (time domain transmission), or TDR/TDT (both TDR and TDT).

Parameters:
<Type> TDR | TDRTdt | TDT
*RST: TDR

Usage: Asynchronous command

TDRT:PULSe:OUT <Channel>

Sets the channel to which the "Out" connector of the pulse source is connected via calibration standard or DUT.

Parameters:
<Channel> C1W1 | C2W1 | C3W1 | C4W1
*RST: C2W1

Usage: Asynchronous command

TDRT:PULSe:REFerence <Channel>

Sets the channel to which the "Ref" connector of the pulse source is connected.

Parameters:
<Channel> C1W1 | C2W1 | C3W1 | C4W1
*RST: C1W1

Usage: Asynchronous command

TDRT:SLENgth <ShortLength>

Sets the electrical length of a user-defined short.

The reset value is the electrical length of the short hat is delivered with the pulse source R&S RTO6-B7.

Parameters:
<ShortLength> Range: 0 to 2E-9
Increment: 1E-12
*RST: 26.5E-12
Default unit: s

Usage: Asynchronous command

TDRT:CALibration:SAVE:NAME <Path>

Sets the file name, file format and path of the TDR/TDT calibration file to be saved. The file format is always `.xml`.

Parameters:
<Path> String with path and file name with extension `.xml`

Usage: Asynchronous command

TDRT:CALibration:SAVE[:EXECute]

Saves the TDR/TDT calibration results to the file that is specified with `TDRT:CALibration:SAVE:NAME`.

Usage: Event
Asynchronous command

24.20.2.2 Measurements

TDRT:STATe....2468
TDRT:CALibration:USECaldata....2468
TDRT:BW....2469
TDRT:DOMain....2469
TDRT:PERMittivity....2469
TDRT:SIGNal<m>STATe....2469
TDRT:SIGNal<m>:SCALe....2470
TDRT:SIGNal<m>:OFFSet....2470
TDRT:SIGNal<m>:POSition....2470
TDRT:SIGNal<m>:DATA:HEADer?....2471
TDRT:SIGNal<m>:DATA:VALues?....2471

TDRT:STATe <State>

Activates or deactivates the TDR/TDT measurements.

Parameters:
<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

TDRT:CALibration:USECaldata <UseCalibrationDat>

Disables or enables the frequency response calibration. Without calibration, the measured signals are only rescaled, and you can see the impact of the frequency response calibration on the time domain waveform.

The command is not relevant when measuring with default values.

Parameters:
<UseCalibrationDat> ON | OFF
*RST: ON

Usage: Asynchronous command

TDRT:BW <Bandwidth>

Sets the bandwidth of the DUT.

Parameters:
<Bandwidth> Range: 2E+9 to 16E+9
Increment: 10
*RST: 4E+9
Default unit: Hz

Usage: Asynchronous command

TDRT:DOMain <OutputDomain>

Sets the domain of the x-axis: time or distance (in m or inch). If you select distance, enter also the permittivity with `TDRT:PERMittivity`.

Parameters:
<OutputDomain> TIME | MDIStance | INDistance
MDIStance = distance in m
IDIStance = distance in inch
*RST: TIME

Usage: Asynchronous command

TDRT:PERMittivity <Permittivity>

Sets the permittivity "ε_r" of the used DUT.

Parameters:
<Permittivity> Range: 1 to 100
Increment: 0.1
*RST: 3.3

Usage: Asynchronous command

TDRT:SIGNal<m>STATe <State>

Enables the indicated TDR/TDT signal.

Suffix:
<m> 1..3
1 = impedance Z (result of TDR measurement)
2 = reflection coefficient ρ (result of TDR measurement)
3 = step response h (result of TDT measurement)

Parameters:

<State> ON | OFF

*RST: OFF

Usage: Asynchronous command

TDRT:SIGNal<m>:SCALe <VerticalScale>

Set the vertical scale of the indicated TDR/TDT signal.

Suffix:

<m> 1..3

See TDRT:SIGNal<m>STATe on page 2469.

Parameters:

<VerticalScale>
Range: 1E-12 to 100E+12
Increment: 1E-3
*RST: 0.05
Default unit: /div

Usage: Asynchronous command

TDRT:SIGNal<m>:OFFSet <Offset>

Set the offset of the indicated TDR/TDT signal.

Suffix:

<m> 1..3

See TDRT:SIGNal<m>STATe on page 2469.

Parameters:

<Offset>
Range: 1E-12 to 100E+12
Increment: 0.01
*RST: 0

Usage: Asynchronous command

TDRT:SIGNal<m>:POSition <Position>

Set the vertical position of the indicated TDR/TDT signal.

Suffix:

<m> 1..3

See TDRT:SIGNal<m>STATe on page 2469.

Parameters:

<Position>
Range: 1E-12 to 100E+12
Increment: 0.02
*RST: 0
Default unit: div

Usage: Asynchronous command

TDRT:SIGNal<m>:DATA:HEADer?

Returns the header of channel waveform data.

Table 24-17: Header data

Position	Meaning	Example
1	XStart in s	-9.477E-008 = - 94,77 ns
2	XStop in s	9.477E-008 = 94,77 ns
3	Record length of one waveform	200000
4	Number of values per sample interval, is 1	1

Suffix:
<m> 1..3
1 = impedance Z (result of TDR measurement)
2 = reflection coefficient ρ (result of TDR measurement)
3 = pulse response h (result of TDT measurement)

Usage: Query only

TDRT:SIGNal<m>:DATA:VALues? [<Offset>], [<Length>]

Returns the data of the waveform points for transmission from the instrument to the controlling computer. The data can be used in MATLAB, for example.

Without parameters, the complete waveform is retrieved. Using the offset and length parameters, data can be retrieved in smaller portions, which makes the command faster.

To set the export format, use `FORMat[:DATA]`.

Suffix:
<m> 1..3
1 = impedance Z (result of TDR measurement)
2 = reflection coefficient ρ (result of TDR measurement)
3 = pulse response h (result of TDT measurement)

Query parameters:
[<Offset>] Number of offset waveform points.
Range: 0 to m. Limit: n + m >= record length

Return values:
[<Length>] Number of waveform points to be retrieved.
Range: 1 to n. Limit: n + m >= record length

Usage: Query only

24.21 I/Q software interface (option R&S RTO6-K11)

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

- I/Q sampling settings....2472
- I/Q data output....2477
- NFC trigger....2478

24.21.1 I/Q sampling settings

To switch on the input channels, use `CHANnel<m>:STATe`.

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

IQ:STATe....2473
IQ:INSRate....2473
CHANnel<m>:IQ:INPType....2473
CHANnel<m>:IQ:INPMode....2473
IQ:RBWidth....2474
IQ:BWIDth?....2474
IQ:SRATe....2474
IQ:RLENgth....2474
CHANnel<m>:IQ:CFRequency....2475
CHANnel<m>:IQ:SBRF....2475
CHANnel<m>:IQ:SBIF....2476
CHANnel<m>:IQ:WARNing[:STATe]?....2476
CHANnel<m>:IQ:WARNing:MESSage?....2476

IQ:STATe <IQMode>

Activates the I/Q mode of the instrument.

Parameters:

<IQMode> ON | OFF

ON: I/Q mode
OFF: normal oscilloscope mode

*RST: OFF

Usage: Asynchronous command

IQ:INSRate <InputSampleRate>

Shows the sample rate of capturing the input data in I/Q mode: 10 Gsample/s.

Return values:

<InputSampleRate> SR10G

Usage: Asynchronous command

CHANnel<m>:IQ:INPType <IQInputType>

Sets the type of the input signal.

Suffix:

<m> 1..4

Parameters:

<IQInputType> REAL | COMPlex

REAL
Real RF signal. One real RF signal requires one input channel, thus up to four real signals can be recorded in parallel.

COMPlex
Complex I/Q signal in baseband or low IF range. One complex input signal requires two input channels. The In-Phase component must be connected to channel 1 or 3, and the Quadrature component must be connected to channel 2 or 4. Thus up to two complex input signals can be recorded in parallel.

*RST: REAL

Usage: Asynchronous command

CHANnel<m>:IQ:INPMode <IQInputMode>

Selects the frequency band of a complex input signal.

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<IQInputMode> BASeband | RFIF

Baseband or intermediate frequency (RFIF)

*RST: BASeband

Usage: Asynchronous command

IQ:RBWidth <RelBw>

Sets the bandwidth factor to define the filter bandwidth.

*Filter BW = Relative BW * Sample rate*

Parameters:

<RelBw> Range: 0.04 to 0.8
Increment: 0.01
*RST: 0.8

Usage: Asynchronous command

IQ:BWIDth?

Returns the filter bandwidth.

See also: `IQ:RBWidth` on page 2474

Return values:

<AbsBw> Range: 40 to 8E+9
*RST: 4E+9
Default unit: Hz

Usage: Query only
Asynchronous command

IQ:SRATe <SampleRate>

Sets the required sample rate of the output I/Q data.

Parameters:

<SampleRate> Range: 1 to 10E+9
Increment: 1
*RST: 10E+9
Default unit: Sa/s

Usage: Asynchronous command

IQ:RLENgth <RecordLength>

Sets the required record length of the output I/Q data. The resulting acquisition time of the I/Q data is:

Acquisition time = Record length / Sample rate

Parameters:

<RecordLength> Range: 1 to 10E+6
Increment: 2
*RST: 100000
Default unit: pts

Usage: Asynchronous command

CHANnel<m>:IQ:CFRequency <IQCarrierFreq>

Sets the carrier frequency of the modulated RF signal or of the complex signal in IF range.

Prerequisites:

- `CHANnel<m>:IQ:INPType` on page 2473 is set to `REAL`
  or:
- `CHANnel<m>:IQ:INPType` on page 2473 is set to `COMPlex` and `CHANnel<m>:IQ:INPMode` on page 2473 is set to `RFIF`

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<IQCarrierFreq> Range: 10E-3 to 10E+9
Increment: 100
*RST: 1E+6
Default unit: Hz

Usage: Asynchronous command

CHANnel<m>:IQ:SBRF <IQSidebandRF>

Defines the frequency position of the RF spectrum in the input signal: normal or inverse. The position is important for correct down-conversion and filtering.

Prerequisite: `CHANnel<m>:IQ:INPType` on page 2473 is set to `REAL`

For details, see "Sideband (real input)" on page 1063

Suffix:

<m> 1..4
Selects the input channel.

Parameters:

<IQSidebandRF> INVerse | NORMal
*RST: NORMal

Usage: Asynchronous command

CHANnel<m>:IQ:SBIF <IQSidebandIF>

Defines the sideband and the frequency position of complex modulated input signal in IF range.

Prerequisites: `CHANnel<m>:IQ:INPType` on page 2473 is set to `COMPlex` and `CHANnel<m>:IQ:INPMode` on page 2473 is set to `RFIF`

For details, see "Sideband (complex IF input)" on page 1064

Suffix:
<m> 1..4
Selects the input channel.

Parameters:
<IQSidebandIF> NUPPer | NLOWer | IUPPer | ILOWer

NUPPer: Upper sideband in normal position
NLOWer: Lower sideband in normal position
IUPPer: Upper sideband in inverse position
ILOWer: Lower sideband in inverse position

*RST: NUPPer

Usage: Asynchronous command

CHANnel<m>:IQ:WARNing[:STATe]?

Returns ON if the sampling setup results in an unfavorable combination.

To get the message, use `CHANnel<m>:IQ:WARNing:MESSage?`.

Suffix:
<m> 1..4

Return values:
<WarningState> ON | OFF

*RST: OFF

Usage: Query only

CHANnel<m>:IQ:WARNing:MESSage?

Returns the message string if `CHANnel<m>:IQ:WARNing[:STATe]?` is true.

Suffix:
<m> 1..4

Return values:
<WarningMessage> The string is:
"Undersampling" if the carrier frequency is too fast.
"CF + BW/2 > DevBW" if bandwidth serttings do not fit. CF is the carrier frequency, BW is the filter bandwidth, and DevBW is the maximum instrument bandwidth.

Usage: Query only

24.21.2 I/Q data output

CHANnel<m>:IQ:DATA[:VALues]?..... 2477
CHANnel<m>:IQ:DATA:HEADer?..... 2477

CHANnel<m>:IQ:DATA[:VALues]?

Returns the recorded I/Q data.

Suffix:

<m> 1..4
Selects the input channel.
In case of a complex input signal that requires two input channels, the results of sources Ch1 and Ch2 are identical, as well as the results of Ch3 and Ch4.

Return values:

<Data> In-phase and and Quadrature floating values in interleaved order.

Example:

```
CHAN1:IQ:DATA?
-9.6296054835E-005, -1.5046258568E-006,
0.0001013283545, 1.5832555391E-006,
-0.00014297277085, -2.233616442E-006,
0.000192677413, 3.0105845781E-006,
-0.00020517286612, -3.2058260331E-006,
-0.002648930531, -4.1546467401E-005,
-0.0028401580639, -4.4135249482E-005,
-0.0028636774514, -4.4877564505E-005, ...
```

Usage: Query only

CHANnel<m>:IQ:DATA:HEADer?

Returns the header of I/Q data.

Table 24-18: Header data

Position	Meaning	Example
1	XStart, acquisition time before trigger, in s	-5E-008 = - 50 ns
2	XStop, acquisition time after trigger, in s	5E-008 = 50 ns
3	Record length of the waveform in Samples	1000
4	Number of values per sample interval. For I/Q data the result is 1.	1

Suffix:

<m> 1..4
Selects the input channel.

Example:

```
CHAN1:IQ:DATA:HEAD?
-5E-008,5E-008,1000,1
```

Usage: Query only

24.21.3 NFC trigger

TRIGger<m>:NFC:TECHnology....2478
TRIGger<m>:NFC:BITRate....2478
TRIGger<m>:NFC:EVENt....2478

TRIGger<m>:NFC:TECHnology <Technology>

Selects the NFC technology, the communication protocol used by the input signal.

Suffix:
<m> 1..3
Event in a trigger sequence: 1 = A-event only

Parameters:
<Technology> NFCA | NFCB | NFCF
*RST: NFCA

Usage: Asynchronous command

TRIGger<m>:NFC:BITRate <Bitrate>

Sets the bit rate of the signal. For NFC-A and NFC-B, the bit rate is always 106 kBit per second. For NFC-F, the bit rate can be either 212 or 424 kBit per second.

Suffix:
<m> 1..3
Event in a trigger sequence: 1 = A-event only

Parameters:
<Bitrate> KBPS106 | KBPS212 | KBPS424
*RST: KBPS106

Usage: Asynchronous command

TRIGger<m>:NFC:EVENt <Event>

Sets the event to be triggered on.

Suffix:
<m> 1..3
Event in a trigger sequence: 1 = A-event only

Parameters:
<Event> ALL | SEN | ALLB | SENB | SOS48 | SOS96

ALL | SEN
ALL_REQ and SENS_REQ commands (polling requests), only with NFC-A technology

ALLB | SENB
ALLB_REQ and SENSB_REQ commands, only with NFC-B technology

SOS48 | SOS96
Start of sequence, 48 bit 96 bit land length, only with NFC-F technology

*RST: SEN

Usage: Asynchronous command

24.22 Jitter analysis and clock data recovery

This chaptes decribes the remote commands of jitter option R&S RTO6-K12/K133 and CDR option R&S RTO6-K13.

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

- Jitter measurements (option R&S RTO6-K12).....2479
- Advanced jitter and noise(option R&S RTO6-K133)/(option R&S RTO6-K134). 2484
- Clock data recovery (software-based, option R&S RTO6-K12).....2510
- Clock data recovery (hardware-based, option R&S RTO6-K13).....2515
- Serial pattern trigger using CDR.....2518
- Eye mask testing.....2519

24.22.1 Jitter measurements (option R&S RTO6-K12)

The following table lists the measurement suffixes and the <MeasType> parameter value with a short description.

For a detailed description, see Chapter 18.1.1, "Jitter measurements", on page 1069.

Table 24-19: Jitter measurements

Meas. suffix	<MeasType> parameter value	Meas. type	Description/Result
37	CCJitter	Cycle-cycle jitter	Difference between the periods of two adjacent cycles. The measurement is based on the period measurement.
38	NCJitter	N-cycle jitter	Difference between the time of two adjacent groups of N cycles (periods) each.
39	CCWidth	Cycle-cycle width	Difference between the pulse width of two adjacent cycles. The measurement is based on the pulse width measurement.
40	CCDutycycle	Cycle-cycle duty cycle	Difference between the duty cycle of two adjacent cycles. The measurement is based on the duty cycle measurement.
41	TIE	Time interval error	Time difference between the slope of the input signal and the slope of a reference signal. The reference signal can be a captured clock waveform, or a clock generated by clock data recovery (CDR, software algorithm or hardware generation).
42	UINTerval	Unit interval	Period of the clock signal. If no clock signal is available, it is recovered by CDR. The period is calculated as the time difference between two consecutive clock edges of the same polarity.
43	DRATe	Data rate	Frequency of the clock signal. If no clock signal is available, it is recovered by CDR. The measurement is based on the unit interval measurement.
44	SKWDelay	Skew delay	Delay between the edges of two interdependent waveforms.
45	SKWPhase	Skew phase	Phase difference between the edges of two waveforms.
1 to 36; 46	Used for amplitude/time measurements (limit checks). The jitter category uses the same limit checks as amplitude/time. See Chapter 24.12.3, "Amplitude/time measurements", on page 1489.		

MEASurement<m>:JITTer:CCSLope....2480
MEASurement<m>:JITTer:PULSe....2481
MEASurement<m>:JITTer:NCYCles....2481
MEASurement<m>:JITTer:CDRMode....2482
MEASurement<m>:JITTer:SOURce<n>:TIESlope....2482
MEASurement<m>:JITTer:DRUNit....2483
MEASurement<m>:JITTer:SKWSlope....2483
MEASurement<m>:JITTer:SKWRelation....2484

MEASurement<m>:JITTer:CCSLope <JitterCCSlope>

Selects the slope at which the periods and thus the jitter is measured.

The command is available for the following measurements: cycle-cycle jitter, N-cycle jitter, and cycle-cycle duty cycle (`MEASurement<m>:MAIN` and/or `MEASurement<m>:ADDitional` are set to measurements CCJitter | NCJitter | CCDutycycle).

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<JitterCCSlope> FIRSt | POSitive | NEGative | EITHer

FIRSt
Measures the period from the first edge that is found, no matter of its direction.

POSitive
Measures the period at positive going edges.

NEGative
Measures the period at negative going edges.

EITHer
Measures the period at both positive and negative going edges. This option is useful, for example, to check the clock stability of a double data rate clock.

*RST: FIRSt

Usage: Asynchronous command

MEASurement<m>:JITTer:PULSe <JittPlsPola>

Sets the polarity of pulses for which the pulse width is measured to obtain the cycle-cycle width and the cycle-cycle duty cycle.

The command is available for the following measurements: cycle-cycle width and cycle-cycle duty cycle (`MEASurement<m>:MAIN` and/or `MEASurement<m>:ADDitional` are set to measurements CCWidth | CCDutycycle).

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<JittPlsPola> POSitive | NEGative

Pulse width of positive or negative pulses is measured, respectively.

*RST: POSitive

Usage: Asynchronous command

MEASurement<m>:JITTer:NCYCles <JitterNofCycles>

Sets the number of periods (cycles) that are accumulated to measure the N-cycle jitter.

See also: Table 18-1

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<JitterNofCycles> Range: 1 to 2147483647
Increment: 1
*RST: 2

Usage: Asynchronous command

Firmware/software: FW 2.60. The algorithm of N-cycle measurement has been changed and `MEASurement:JITTer:COFFset` has been removed.

MEASurement<m>:JITTer:CDRMode <JitterCDRMode>

Defines the origin of the clock signal - wether a real clock signal or a clock generated using one of the CDR methods.

The command is available for the following measurements: time-interval error, unit interval and data rate (`MEASurement<m>:MAIN` and/or `MEASurement<m>:ADDitional` are set to measurements TIE | UINTerval | DRATe).

Suffix:
<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:
<JitterCDRMode> SIGNal | SW1 | SW2 | HW

SIGNal
The clock is a real clock signal.

SW1 | SW2
The clock is generated by a software algorithm. You can define two software algorithms for CDR. To configure the CDR, use the `CDR:SOFTware` commands.

HW
The clock is generated from the source signal by the integrated CDR circuitry. The HW CDR must be active to acquire the jitter data (trigger type = CDR or serial pattern with CDR)

*RST: SW1

Usage: Asynchronous command

MEASurement<m>:JITTer:SOURce<n>:TIESlope <JitterTIESlope>

Sets the clock and data edges that are used for measurements. Clock or data is defined by the SOURce suffix <n>.

The command is available for the following measurements: time-interval error, unit interval and data rate (`MEASurement<m>:MAIN` and/or `MEASurement<m>:ADDitional` are set to measurements TIE | UINTerval | DRATe.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

<n> 1..2
1 = data slope (only relevant for time interval error measurements with explicit clock signal)
2 = clock slope

Parameters:

<JitterTIESlope> POSitive | NEGative | EITHer

POSitive
The positive clock slope can be used, for example, for single data rate (SDR) signals with bit start at the positive clock edge.

NEGative
The negative clock slope can be used, for example, for SDR signals with bit start at the negative clock edge.

EITHer
For clock edges, this option can be used for double data rate (DDR) signals.
For data edges, it is the most common setting.

*RST: EITHer

Usage: Asynchronous command

MEASurement<m>:JITTer:DRUNit <JittDatRateUnt>

Sets the unit for data rate measurements.

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<JittDatRateUnt> BIT_S | Hz

*RST: BIT_S

Usage: Asynchronous command

MEASurement<m>:JITTer:SKWSlope <JitterSkewSlope>

Sets the edge of the first waveform from which the skew delay or phase is measured: positive, negative or both.

The command is available for the following measurement: skew delay and skew phase (`MEASurement<m>:MAIN` and/or `MEASurement<m>:ADDitional` are set to measurements SKWDelay | SKWPhase).

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<JitterSkewSlope> POSitive | NEGative | EITHer

*RST: POSitive

Usage: Asynchronous command

MEASurement<m>:JITTer:SKWRelation <JittskwRelt>

Sets the edge of the second waveform relative to the first waveform.

The command is available for the following measurements: skew delay and skew phase (`MEASurement<m>:MAIN` and/or `MEASurement<m>:ADDitional` are set to measurements SKWDelay | SKWPhase).

Suffix:

<m> 1..10
See "Selection of the measurement group: MEASurement<m>" on page 1479.

Parameters:

<JittskwRelt> MATChing | INVerse

MATChing
Measures from positive to positive edge or from negative to negative edge.

INVerse
Measures from positive to negative edge or from negative to positive edge.

*RST: MATChing

Usage: Asynchronous command

24.22.2 Advanced jitter and noise(option R&S RTO6-K133)/(option R&S RTO6-K134)

This chapter contains all remote commands to set up the advanced jitter and advanced noise analysis.

Selection of the advanced jitter measurement: ADVJitter<m>

With R&S RTO6, you can configure up to two simultaneous advanced jitter measurements. In manual operation, they are represented by subtabs "JA1" and "JA2" in the

"Advanced Jitter" dialog box. For remote operation, they are indicated by the suffix `ADVJitter<m>`.

Remote control: ADVJitter suffix <m>	Manual operation: "JA" subtab
1 to 2 correspond to	"JA1" to "JA2"

- Signal....2485
- Results....2492
- Advanced settings....2507

24.22.2.1 Signal

ADVJitter<m>:SIGNal:TYPE....2485
ADVJitter<m>:SIGNal:SOURce....2486
ADVJitter<m>:SIGNal:QUICkmeas....2486
ADVJitter<m>:SIGNal:BITRate....2486
ADVJitter<m>:SIGNal:CDRMode....2486
ADVJitter<m>:SIGNal:REFSource....2486
ADVJitter<m>:SIGNal:CLKSource....2487
ADVJitter<m>:SIGNal:EDGE....2487
ADVJitter<m>:SIGNal:CLEStimation....2487
ADVJitter<m>:SIGNal:CLFFactor....2487
ADVJitter<m>:SIGNal:CFEStimation....2488
ADVJitter<m>:SIGNal:CLFRequency....2488
ADVJitter<m>:SIGNal:CLOFfset....2488
ADVJitter<m>:SIGNal:DATRate?....2488
ADVJitter<m>:SIGNal:SAMTime....2489
ADVJitter<m>:SIGNal:GATE:COUPling....2489
ADVJitter<m>:SIGNal:GATE:CURSor....2489
ADVJitter<m>:SIGNal:GATE:RANGe:ABSolute:STARt....2489
ADVJitter<m>:SIGNal:GATE:RANGe:ABSolute:STOP....2490
ADVJitter<m>:SIGNal:GATE:RANGe:MODE....2490
ADVJitter<m>:SIGNal:GATE:RANGe:RELative:STARt....2490
ADVJitter<m>:SIGNal:GATE:RANGe:RELative:STOP....2490
ADVJitter<m>:SIGNal:GATE:ZDIagram....2491
ADVJitter<m>:SIGNal:GATE:STATe....2491
ADVJitter<m>:SIGNal:SERStandard....2491

ADVJitter<m>:SIGNal:TYPE <SignalType>

Select, if the analyzed signal is a data or clock signal.

Suffix:
<m> 1..2

Parameters:
<SignalType> CLOCk | DATNrz

*RST: DATNrz

ADVJitter<m>:SIGNal:SOURce <SourceType>

Selects the source for the advanced jitter analysis.

Suffix:
<m> 1..2

Parameters:
<SourceType> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

ADVJitter<m>:SIGNal:QUICkmeas

Starts a quick advanced jitter analysis.

Suffix:
<m> 1..2

Usage: Event

ADVJitter<m>:SIGNal:BITRate <NominalBitRate>

Suffix:
<m> 1..2

Parameters:
<NominalBitRate> Range: 1000 to 10E+9
Increment: 10
*RST: 1E+9
Default unit: bps

ADVJitter<m>:SIGNal:CDRMode <MethodType>

Selects the reference source for the CDR signal, if `ADVJitter<m>:SIGNal:REFSource` is set to `CDR`.

Suffix:
<m> 1..2

Parameters:
<MethodType> SW1 | SW2
*RST: SW1

ADVJitter<m>:SIGNal:REFSource <SourceType>

Selects if an explicit clock or a CDR signal is used as a reference signal.

Suffix:
<m> 1..2

Parameters:
<SourceType> CLOCk | CDR
*RST: CDR

ADVJitter<m>:SIGNal:CLKSource <SourceType>

Selects the reference source for the clock signal, if `ADVJitter<m>:SIGNal:REFSource` is set to `CLOCk`.

Suffix:
<m> 1..2

Parameters:
<SourceType> C1W1 | C1W2 | C1W3 | C2W1 | C2W2 | C2W3 | C3W1 | C3W2 | C3W3 | C4W1 | C4W2 | C4W3 | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 | R1 | R2 | R3 | R4 | DIFF1 | DIFF2 | COMMON1 | COMMON2

ADVJitter<m>:SIGNal:EDGE <EdgeType>

The clock edge sets the point in time when the state of the data signal is checked.

Suffix:
<m> 1..2

Parameters:
<EdgeType> POSitive | NEGative | EITHer
*RST: EITHer

ADVJitter<m>:SIGNal:CLEStimation

Executes an estimation of the signal for the clock frequency.

Suffix:
<m> 1..2

Usage: Event
Asynchronous command

ADVJitter<m>:SIGNal:CLFFactor <Factor>

Sets a clock multiplier factor.

Suffix:
<m> 1..2

Parameters:
<Factor> Range: 1 to 100
Increment: 1
*RST: 1

ADVJitter<m>:SIGNal:CFEStimation

Executes an estimation of the signal for the clock offset.

Suffix:
<m> 1..2

Usage: Event
Asynchronous command

ADVJitter<m>:SIGNal:CLFRequency <Frequency>

Sets the clock frequency for the explicit clock signal.

Suffix:
<m> 1..2

Parameters:
<Frequency>
Range: 1000 to 10E+9
Increment: 10
*RST: 1E+9
Default unit: Hz

ADVJitter<m>:SIGNal:CLOFfset <Offset>

Sets a clock offset for the explicit clock signal.

Suffix:
<m> 1..2

Parameters:
<Offset>
Range: -1000 to 1000
Increment: 0.01
*RST: 0
Default unit: UI

ADVJitter<m>:SIGNal:DATRate?

Displays the calculated data rate of the explicit clock signal.

Suffix:
<m> 1..2

Return values:
<DatarateValue>
Range: 1000 to 10E+9
Increment: 10
*RST: 1E+9
Default unit: bps

Usage: Query only

ADVJitter<m>:SIGNal:SAMTime <SamplingTime>

Sets a sampling time for the explicit clock signal, an offset for the clock edge in relation to the bit start. The clock edge sets the beginning of the unit interval. The sampling time is a number between 0 and 1. Value 0 sets the clock edge to the beginning of the bit period; value 0.5 sets the clock edge to the middle of the bit period.

Suffix:
<m> 1..2

Parameters:
<SamplingTime> Range: 0 to 1
Increment: 0.01
*RST: 0.5
Default unit: UI

ADVJitter<m>:SIGNal:GATE:COUPling <ModeType>

Sets if the gate is coupled to the zoom or cursor.

Suffix:
<m> 1..2

Parameters:
<ModeType> NONE | CURSor | ZOOM
*RST: NONE

ADVJitter<m>:SIGNal:GATE:CURSor <CursorType>

Sets the cursor the gate is coupled to, if `ADVJitter<m>:SIGNal:GATE:COUPling` is set to `CURSor`.

Suffix:
<m> 1..2

Parameters:
<CursorType> CURSOR1 | CURSor1 | CURSOR2 | CURSor2 | CURSOR3 | CURSor3 | CURSOR4 | CURSor4
*RST: CURSOR1

ADVJitter<m>:SIGNal:GATE:RANGe:ABSolute:STARt <StartValue>

Sets the start time of the gate area in percent, if `ADVJitter<m>:SIGNal:GATE:RANGe:MODE` is set to `ABS`.

Suffix:
<m> 1..2

Parameters:
<StartValue> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0

ADVJitter<m>:SIGNal:GATE:RANGe:ABSolute:STOP <StopValue>

Sets the stop time of the gate area in percent, if `ADVJitter<m>:SIGNal:GATE:RANGe:MODE` is set to `ABS`.

Suffix:
<m> 1..2

Parameters:
<StopValue> Range: -100E+24 to 100E+24
Increment: 0.01
*RST: 0.01

ADVJitter<m>:SIGNal:GATE:RANGe:MODE <ModeType>

Selects if the gate area is defined in relative values (percent) or absolute.

Suffix:
<m> 1..2

Parameters:
<ModeType> ABS | REL
*RST: ABS

ADVJitter<m>:SIGNal:GATE:RANGe:RELative:STARt <StartValue>

Sets the relative start of the gate area in percent, if `ADVJitter<m>:SIGNal:GATE:RANGe:MODE` is set to `REL`.

Suffix:
<m> 1..2

Parameters:
<StartValue> Range: 0 to 100
Increment: 0.1
*RST: 0
Default unit: %

ADVJitter<m>:SIGNal:GATE:RANGe:RELative:STOP <StopValue>

Sets the relative stop of the gate area in percent, if `ADVJitter<m>:SIGNal:GATE:RANGe:MODE` is set to `REL`.

Suffix:
<m> 1..2

Parameters:
<StopValue> Range: 0 to 100
Increment: 0.1
*RST: 100
Default unit: %

ADVJitter<m>:SIGNal:GATE:ZDIagram <ZoomType>

Sets the zoom the gate is coupled to, if `ADVJitter<m>:SIGNal:GATE:COUPling` is set to `ZOOM`.

Suffix:
<m> 1..2

Parameters:
<ZoomType>

ADVJitter<m>:SIGNal:GATE:STATe <State>

Enables the gating setup for the advanced jitter analysis.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: OFF

ADVJitter<m>:SIGNal:SERStandard <SerStandardType>

Sets the serial standard. Set `CUSTOM` to adjust the bit rate and the bandwidth manually.

The serial standards are available only if the instrument is able to support its maximum nominal bit rate.

Suffix:
<m> 1..2

Parameters:
<SerStandardType> CUSTOM | USB20 | USB31GEN1 | USB31GEN2 | DISPPORT11 | DISPPORT12 | FIBCHAN1 | FIBCHAN2 | FIBCHAN4 | FIBCHAN8 | FIBCHAN16 | HDMI12 | HDMI14 | HDMI20 | PCIGEN1 | PCIGEN2 | PCIGEN3 | PCIGEN4 | SATAGEN1500 | SATAGEN11667 | SATAGEN1 | SATAGEN2500 | SATAGEN21667 | SATAGEN2 | SATAGEN3500 | SATAGEN31667 | SATAGEN3 | SASGEN1 | SASGEN2 | SASGEN3 | TB10G | XAUI | CUSTOM | USB20 | USB31GEN1 | DISPPORT11 | DISPPORT12 | FIBCHAN1 | FIBCHAN2 | FIBCHAN4 | HDMI12 | HDMI14 | HDMI20 | PCIGEN1 | PCIGEN2 | SATAGEN1500 | SATAGEN11667 | SATAGEN1 | SATAGEN2500 | SATAGEN21667 | SATAGEN2 | SATAGEN3500 | SATAGEN31667 | SATAGEN3 | SASGEN1 | SASGEN2 | XAUI
*RST: CUSTOM

Usage: Asynchronous command

24.22.2.2 Results

For remote operation, the component is indicated by the suffix `COMPonents<n>`.

Table 24-20: Advanced jitter components

Suffix <n>	Component	Description/Result
Basic components		
1	Symbol rate	Number of data symbols per second of the analyzed signal. It is equivalent to the baud rate. For binary signals, the signal rate is equal to the bit rate.
2	Symbol duration	The length of time to transfer one signal symbol. Equals the inverse value of the symbol rate.
3	Event Count	Number of analyzed data transitions.
Jitter components (option R&S RTO6-K133)		
4	TJ@BER	The eye closure at the user-defined target BER.
5	RJ+(O)BUJ	Consists of any influence that cannot be predicted with model parameters. Usually, this jitter component follows a Gaussian probability distribution. Other Bounded Uncorrelated Jitter (OBUJ) is included here. OBUJ refers to any jitter that is bounded, uncorrelated time interval error (TIE) distribution, for example from unobserved data signals.
6	DJ (δ-δ)	Industry standard approximation to the deterministic jitter measurement, calculated by fitting a model of two overlapping, equally large Gaussian distributions to the observed DJ histogram. The calculated value is the time difference between the two distributions center points.
7	TJ	Measures the total Time Interval Error (TIE) of the input signal. It is defined as the time between each transition of the input signal and its corresponding reference clock transition.
8	DJ	The reproducible portion of the overall TIE of the input signal. It consists of the Data Dependent Jitter (DDJ) and Periodic Jitter (PJ) components.
9	DDJ	Consists of Inter-symbol Interference (ISI) and Duty Cycle Distortion (DCD). It depends on the symbol sequence. ISI is an effect commonly caused by limited signal bandwidth in the transmitter and transmission line, which forces influence from past symbols onto future symbols. DCD is a mismatch between rising and falling signal transitions due to reference level mismatch or other imbalances between transitions.
10	DCD	Measure of imbalance between rising and falling data transitions. It is defined as the time difference between the average rising and average falling edge.
11	PJ	Caused by crosstalk from periodic signal sources to the transmitter or transmission line. It represents the harmonic portion of the overall time interval error (TIE).
12	ISI	The time between the earliest and the latest signal transition of a single transition type, i.e.rising and falling edges. The measurement result is the largest value across all transition types.

Suffix <n>	Component	Description/Result
13	DDJ +PJ	Composite impact of all unknown but deterministic portions of the overall time interval error of the input signal. It consists of any disturbance that comes from a deterministic source but cannot be reproduced from the input signal. Often it is caused by crosstalk from other data signals.
14	(O)BUJ	Composite impact of all unknown but deterministic portions of the overall time interval error of the input signal. It consists of any disturbance that comes from a deterministic source but cannot be reproduced from the input signal. Often it is caused by crosstalk from other data signals.
15	(O)BUJ (δ-δ)	Other bounded uncorrelated Dual-Dirac jitter (O)BUJ (δ- δ) is an industry standard approximation to the other bounded uncorrelated jitter measurement. It is calculated by fitting a model of two overlapping, equally large Gaussian distributions to the observed RJ+(O)BUJ histogram. The calculated value is the amplitude difference between the two distributions center points.
16	RJ	Consists of any influence that cannot be predicted with model parameters. Usually, this jitter component follows a Gaussian probability distribution. The algorithm computes the standard deviation and the peak-peak value of the TIE caused by RJ within the analyzed waveform.
Noise components (option R&S RTO6-K134)		
17	EH@BER	Returns the vertical eye opening at the user-defined BER.
18	RN+(O)BUN	Random noise plus (other) bounded uncorrelated noise or (other) bounded uncorrelated noise is the combined effect of both noise types. If periodic components are activated RN + OBUN are displayed, otherwise RN + BUN. RN + (O)BUN is typically used, if track or spectrum plots are desired. Both plots are not available for RN only or (O)BUN only components as the time dependencies of these noise types are unknown. RN + (O)BUN spectrum is a useful approximation of the RN spectrum for negligible small (O)BUN.
20	TN	Measures the total level error of the input signal. It is defined as the amplitude deviation between the amplitude of the input signal at the decision time point and its corresponding reference level.
21	DN	The reproducible portion of the overall level error of the input signal. It consists of the data-dependent noise, the periodic noise components and the other bounded uncorrelated jitter components.
22	DDN	Consists of inter-symbol interface noise and level distortion. It depends on the symbol sequence. ISIN is an effect whereby past symbols have an influence on future symbols. LD is a mismatch between rising and falling signal transitions due to reference level mismatch or other imbalances between transitions.
23	LD	A measure of imbalance between different symbol levels. It is defined as the difference between the average level errors of the symbols one and zero.
24	PN	Periodic noise is typically caused by crosstalk from periodic signal sources to the transmitter or transmission line. It represents the harmonic portion of the overall level error.
25	ISIN	The level difference between the highest and the lowest level error of a symbol, i.e. one or zero.The largest value across all symbols is the measurement result.

Suffix <n>	Component	Description/Result
26	DDN + PN	Data-dependent noise plus periodic noise is the combined effect of both jitter types. Note that DDN+PN equals deterministic noise minus other bounded uncorrelated noise. DDN+PN spectrum is a useful approximation of the deterministic noise spectrum fot negligible small other bounded uncorrelated noise.
27	(O)BUN	Composite impact of all unknown but deterministic portions of the overall time interval error of the input signal. It consists of any disturbance that comes from a deterministic source but cannot be reproduced from the input signal. Often it is caused by crosstalk from other data signals.
28	(O)BUN (δ-δ	Other bounded uncorrelated Dual-Dirac noise (O)BUN (δ- δ) is an industry standard approximation to the other bounded uncorrelated jitter measurement. It is calculated by fitting a model of two overlapping, equally large Gaussian distributions to the observed RN+(O)BUN histogram. The calculated value is the amplitude difference between the two distributions center points.
29	RN	Consists of any influence that cannot be predicted with model parameters. Usually, this noise component follows a Gaussian probability distribution. The algorithm computes the standard deviation and the peak-peak value of the level error (LE) caused by RN within the analyzed waveform.

General

ADVJitter<m>:RESult:ENABle....2494
ADVJitter<m>:DCOMposition:COMPonents<n>:ENABle....2495
ADVJitter<m>:RESult:STATistics....2495
ADVJitter<m>:RESult:STEPresponse....2495
ADVJitter<m>:RESult:COMPonents<n>:HISTogram....2495
ADVJitter<m>:RESult:COMPonents<n>:SPECtrum....2496
ADVJitter<m>:RESult:COMPonents<n>:TRACk....2496
ADVJitter<m>:RESult:UNIT....2496
ADVJitter<m>:RESult:COMPonents<n>:STATistics:MAXPeakpeak?....2496
ADVJitter<m>:RESult:COMPonents<n>:STATistics:MAXStddev?....2496
ADVJitter<m>:RESult:COMPonents<n>:STATistics[:MINStddev]?....2496
ADVJitter<m>:RESult:COMPonents<n>:STATistics:MINPeakpeak?....2496
ADVJitter<m>:RESult:COMPonents<n>:STATistics:PEAKpeak?....2496
ADVJitter<m>:RESult:COMPonents<n>:STATistics:WFMCount?....2497
ADVJitter<m>:RESult:COMPonents<n>:STATistics:STDDev?....2497
ADVJitter<m>:RESult:PERiodic<n>:COUNt?....2497
ADVJitter<m>:RESult:PERiodic<n>:DIRection?....2497
ADVJitter<m>:RESult:PERiodic<n>:VALue?....2498
ADVJitter<m>:RESult:PERiodic<n>[:FREQuency]?....2498

ADVJitter<m>:RESult:ENABle <State>

Enables the advanced jitter analysis. It displays all enabled diagrams and displays a table with the results for the component measurements.

Suffix:
<m> 1..2

Parameters:

<State> ON | OFF

*RST: OFF

ADVJitter<m>:DCOMposition:COMPonents<n>:ENABle <State>

Enables the measurement for the specified jitter component.

Suffix:

<m> 1..2

<n> 1..29
See Table 24-20

Parameters:

<State> ON | OFF

*RST: OFF

ADVJitter<m>:RESult:STATistics <State>

Enables statistics measurements for the advanced jitter analysis.

Suffix:

<m> 1..2

Parameters:

<State> ON | OFF

*RST: OFF

ADVJitter<m>:RESult:STEPresponse <State>

Enables a step response diagram for the advanced jitter measurements..

Suffix:

<m> 1..2

Parameters:

<State> ON | OFF

*RST: OFF

ADVJitter<m>:RESult:COMPonents<n>:HISTogram <State>

Enables a histogram for the selected measurement component.

Suffix:

<m> 1..2

<n> 1..29
See Table 24-20

Parameters:

<State> ON | OFF

*RST: OFF

ADVJitter<m>:RESult:COMPonents<n>:SPECtrum <State>

Enables a spectrum diagram for the selected measurement component.

Suffix:

<m> 1..2

<n> 1..29
See Table 24-20

Parameters:

<State> ON | OFF

*RST: OFF

ADVJitter<m>:RESult:COMPonents<n>:TRACk <State>

Enables a track for the advanced jitter measurements.

Suffix:

<m> 1..2

<n> 1..29
See Table 24-20

Parameters:

<State> ON | OFF

*RST: OFF

ADVJitter<m>:RESult:UNIT <UnitType>

Selects between the absolute and relative unit for the displayed results.

Suffix:

<m> 1..2

Parameters:

<UnitType> ABS | REL

*RST: ABS

ADVJitter<m>:RESult:COMPonents<n>:STATistics:MAXPeakpeak? <PeakValue>
ADVJitter<m>:RESult:COMPonents<n>:STATistics:MAXStddev? <StdDevValue>
ADVJitter<m>:RESult:COMPonents<n>:STATistics[:MINStddev]? <StdDevValue>
ADVJitter<m>:RESult:COMPonents<n>:STATistics:MINPeakpeak? <PeakValue>
ADVJitter<m>:RESult:COMPonents<n>:STATistics:PEAKpeak?

ADVJitter<m>:RESult:COMPonents<n>:STATistics:WFMCount? <CountValue>
ADVJitter<m>:RESult:COMPonents<n>:STATistics:STDDev?

Return the statistic results of the specified measurement.

- `MAXPeakpeak`: absolute maximum value of the component.
- `MAXStddev`: standard deviation of the maximum value.
- `MINPeakpeak`: absolute minimum value of the component.
- `MINStddev`: standard deviation of the minimum value.
- `PEAKpeak`: peak to peak value of the long-term measurement results.
- `WFMCount`: number of waveforms (acquisitions) the measurement is based on.
- `STDDev`: standard deviation of the long-term measurement results.

Suffix:

<m>	1..2
<n>	1..29 See Table 24-20

Return values:

<StdDev>	Range:	-100E+24 to 100E+24
	Increment:	100E-12
	*RST:	0

Usage: Query only

ADVJitter<m>:RESult:PERiodic<n>:COUNt?

Returns the number of periodic components.

Suffix:

<m>	1..2
<n>	* Not relevant

Return values:
<Count>

Usage: Query only

ADVJitter<m>:RESult:PERiodic<n>:DIRection?

Returns the direction of the selected periodic component.

Suffix:

<m>	1..2
<n>	* Number of periodic components.

Return values:

<CompDirection> UNDefined | VERTical | HOEXtrinsic | HOINtrinsic

`HOEXtrinsic`: horizontal extrinsic
`HOINtrinsic`: horizontal intrinsic

*RST: UNDefined

Usage: Query only

ADVJitter<m>:RESult:PERiodic<n>:VALue?

Returns the value of the selected periodic component.

Suffix:

<m> 1..2

<n> *
Number of periodic components.

Return values:

<CompValue> Range: -1000 to 1000
Increment: 0
*RST: 0

Usage: Query only

ADVJitter<m>:RESult:PERiodic<n>[:FREQuency]?

Returns the frequency of the selected periodic component.

Suffix:

<m> 1..2

<n> *
Number of periodic components.

Return values:

<FreqValue> Range: 0 to 1E+15
Increment: 0
*RST: 0
Default unit: Hz

Usage: Query only

Measurements

ADVJitter<m>:DCOMposition:CLEBasicmeas 2499
ADVJitter<m>:DCOMposition:SELBasicmeas 2499
ADVJitter<m>:DCOMposition:CLEJitcomp 2499
ADVJitter<m>:DCOMposition:SELJitcomp 2499
ADVJitter<m>:DCOMposition:CLENoisecomp 2499
ADVJitter<m>:DCOMposition:SELNoisecomp 2499
ADVJitter<m>:RESult:TRANsitions 2500
ADVJitter<m>:RESult:LEVels 2500

ADVJitter<m>:DCOMposition:CLEBasicmeas

Clears all selected basic jitter components.

Suffix:
<m> 1..2

Usage: Event
Asynchronous command

ADVJitter<m>:DCOMposition:SELBasicmeas

Selects all basic jitter components.

Suffix:
<m> 1..2

Usage: Event
Asynchronous command

ADVJitter<m>:DCOMposition:CLEJitcomp

Clears all selected jitter components.

Suffix:
<m> 1..2

Usage: Event
Asynchronous command

ADVJitter<m>:DCOMposition:SELJitcomp

Selects all jitter components.

Suffix:
<m> 1..2

Usage: Event
Asynchronous command

ADVJitter<m>:DCOMposition:CLENoisecomp

Clears all selected noise components.

Suffix:
<m> 1..2

Usage: Event
Asynchronous command

ADVJitter<m>:DCOMposition:SELNoisecomp

Selects all noise components.

Suffix:
<m> 1..2

Usage: Event
Asynchronous command

ADVJitter<m>:RESult:TRANsitions <TransitionType>

Sets the measurements for which transitions are displayed in the diagram.

Suffix:
<m> 1..2

Parameters:
<TransitionType> RISing | FALLing | BOTH
*RST: BOTH

ADVJitter<m>:RESult:LEVels <LevelType>

Sets which levels are considered for the display of the diagrams, rising, falling or both. This setting is mostly relevant for histograms.

Suffix:
<m> 1..2

Parameters:
<LevelType> HIGH | LOW | BOTH
*RST: BOTH

Bathtub

ADVJitter<m>:DCOMposition:TARBer....2500
ADVJitter<m>:RESult:BATHtub....2501
ADVJitter<m>:RESult:TYPBathtub....2501
ADVJitter<m>:RESult:PJ....2501
ADVJitter<m>:RESult:OBUJ....2501
ADVJitter<m>:RESult:NBAThtub....2502
ADVJitter<m>:RESult:PN....2502
ADVJitter<m>:RESult:OBUN....2502

ADVJitter<m>:DCOMposition:TARBer <TargetBtErrRate>

Sets the target bit error rate for total jitter at BER components.

Suffix:
<m> 1..2

Parameters:
<TargetBtErrRate> Range: 1 to 32
Increment: 1
*RST: 12

ADVJitter<m>:RESult:BATHtub <State>

Enables a bathtub plot for the advanced jitter measurements.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: OFF

ADVJitter<m>:RESult:TYPBathtub <BathtubType>

Selects if you want to display the bathtub results versus total jitter (TJ) or data-dependent jitter (DDJ) and random jitter (RJ) components.

Suffix:
<m> 1..2

Parameters:
<BathtubType> DDJ | DDJP | DDJB | DDJO | TJ | DDJRj
`TJ`: Total jitter
`DDJRJ`: Data-dependent jitter and random jitter.
*RST: TJ

ADVJitter<m>:RESult:PJ <State>

If enabled, the periodic jitter components are considered for the jitter bathtub.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: ON

ADVJitter<m>:RESult:OBUJ <State>

If enabled, the other bounded uncorrelated jitter components are considered for the jitter bathtub.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: ON

ADVJitter<m>:RESult:NBAThtub <State>

Enables a noise bathtub plot, a graph of BER in a log scale versus data-dependent jitter (DDN).

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: OFF

ADVJitter<m>:RESult:PN <State>

If enabled, the periodic noise components are considered for the noise bathtub.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: ON

ADVJitter<m>:RESult:OBUN <State>

If enabled, the other bounded uncorrelated noise components are considered for the noise bathtub.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: ON

Eye

ADVJitter<m>:RESult:PHORizontal 2502
ADVJitter<m>:RESult:PVERtical 2503
ADVJitter<m>:RESult:SYNTheticeye 2503
ADVJitter<m>:RESult:TYPSynteye 2503

ADVJitter<m>:RESult:PHORizontal <State>

If enabled, adds horizontal periodic components to the data-dependent synthetic eye.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: OFF

ADVJitter<m>:RESult:PVERtical <State>

If enabled, adds vertical periodic components to the data-dependent synthetic eye.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: OFF

ADVJitter<m>:RESult:SYNTheticeye <State>

Selects if you want to display synthetic eye results only for the deterministic jitter (DJ) components or the data-dependent jitter (DDJ) components.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: OFF

ADVJitter<m>:RESult:TYPSynteye <SyntheticEyeTyp>

Selects if you want to display synthetic eye results only for the deterministic jitter (DDP) components or the data-dependent jitter (DD) components.

Suffix:
<m> 1..2

Parameters:
<SyntheticEyeTyp> DD | DDPH | DDPV | DDP
*RST: DDP

Diagram

ADVJitter<m>:RESult:VERTical:HBINs....2504
ADVJitter<m>:RESult:VERTical:CONTinuous....2504
ADVJitter<m>:RESult:VERTical:BMTYpe....2504
ADVJitter<m>:RESult:VERTical:BHContinuous....2504
ADVJitter<m>:RESult:VERTical:JSContinuous....2505
ADVJitter<m>:RESult:VERTical:NHContinuous....2505
ADVJitter<m>:RESult:VERTical:NSContinuous....2505
ADVJitter<m>:RESult:COMPonents<n>:SCALe....2505
ADVJitter<m>:RESult:COMPonents<n>:OFFSet....2506
ADVJitter<m>:DCOMposition:SPJTable....2506
ADVJitter<m>:DCOMposition:SPJLabels....2507
ADVJitter<m>:DCOMposition:SPNLabels....2507
ADVJitter<m>:DCOMposition:JINTerleaved....2507

ADVJitter<m>:RESult:VERTical:HBINs <MaxBinNumber>

Sets the number of bins, the number of data intervals, of the histogram.

Suffix:
<m> 1..2

Parameters:
<MaxBinNumber> Range: 2 to 1000
Increment: 10
*RST: 200

ADVJitter<m>:RESult:VERTical:CONTinuous <State>

If enabled, the measurement scale is set automatically.

If disabled you can manually set a `ADVJitter<m>:RESult:COMPonents<n>:SCALe` and a `ADVJitter<m>:RESult:COMPonents<n>:OFFSet`.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: ON

ADVJitter<m>:RESult:VERTical:BMTYpe <MeasType>

Selects the basic measurement type to apply the vertical scale to.

Suffix:
<m> 1..2

Parameters:
<MeasType> SBRate | SBDuration
`SBRate`: Symbol rate
`SBDuration`: Symbol duration
*RST: SBRate

ADVJitter<m>:RESult:VERTical:BHContinuous <State>

If enabled, the measurement scale of the basic jitter measurement histogram and track is set automatically.

If disabled you can manually set a `ADVJitter<m>:RESult:COMPonents<n>:SCALe` and a `ADVJitter<m>:RESult:COMPonents<n>:OFFSet`.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: ON

ADVJitter<m>:RESult:VERTical:JSContinuous <State>

If enabled, the measurement scale of the jitter spectrum is set automatically.

If disabled you can manually set a `ADVJitter<m>:RESult:COMPonents<n>:SCALe` and a `ADVJitter<m>:RESult:COMPonents<n>:OFFSet`.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: ON

ADVJitter<m>:RESult:VERTical:NHContinuous <State>

If enabled, the measurement scale of the noise histogram and track is set automatically.

If disabled you can manually set a `ADVJitter<m>:RESult:COMPonents<n>:SCALe` and a `ADVJitter<m>:RESult:COMPonents<n>:OFFSet`.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: ON

ADVJitter<m>:RESult:VERTical:NSContinuous <State>

If enabled, the measurement scale for the noise spectrum is set automatically.

If disabled you can manually set a `ADVJitter<m>:RESult:COMPonents<n>:SCALe` and a `ADVJitter<m>:RESult:COMPonents<n>:OFFSet`.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: ON

ADVJitter<m>:RESult:COMPonents<n>:SCALe <Scale>

Sets the scale for the selected component.

You can set separate scale values for:

- `n=1` (symbol rate)
- `n= 2` (symbol duration)
- `n= 4...16`: sets only one offset value that is relevant for all advanced jitter components.
- `n= 17...29`: sets only one offset value that is relevant for all advanced noise components.

Suffix:

<m>	1..2
<n>	1..29 See Table 24-20

Parameters:

<Scale>	Range:	1E-15 to 1E+15
	Increment:	10E-6
	*RST:	0.5
	Default unit:	s/div

ADVJitter<m>:RESult:COMPonents<n>:OFFSet <Offset>

Sets the offset for the selected component.

You can set separate offset values for:

- `n=1` (symbol rate)
- `n= 2` (symbol duration)
- `n= 4...16`: sets only one offset value that is relevant for all advanced jitter components.
- `n= 17...29`: sets only one offset value that is relevant for all advanced noise components.

Suffix:

<m>	1..2
<n>	1..29 See Table 24-20

Parameters:

<Offset>	Range:	-1E+15 to 1E+15
	Increment:	1E-6
	*RST:	0
	Default unit:	s

ADVJitter<m>:DCOMposition:SPJTable <State>

Enables the display of the periodic jitter components table.

Suffix:

<m>	1..2

Parameters:
<State> ON | OFF
*RST: ON

ADVJitter<m>:DCOMposition:SPJLabels <State>

Enables the display of the periodic jitter components labels.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: OFF

ADVJitter<m>:DCOMposition:SPNLabels <State>

Enables the display of the periodic noise components labels.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: OFF

ADVJitter<m>:DCOMposition:JINTerleaved <State>

Displays interleaved histogram values.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: OFF

24.22.2.3 Advanced settings

ADVJitter<m>:DCOMposition:MRESplength? 2508
ADVJitter<m>:DCOMposition:RLEStimate 2508
ADVJitter<m>:DCOMposition:POSStep 2508
ADVJitter<m>:DCOMposition:RESPlength 2508
ADVJitter<m>:DCOMposition:ENP 2509
ADVJitter<m>:DCOMposition:NHORizontal 2509
ADVJitter<m>:DCOMposition:NVERtical 2509
ADVJitter<m>:DCOMposition:ENOBu 2509

ADVJitter<m>:DCOMposition:MRESplength?

Queries the estimated maximum step length response.

Suffix:
<m> 1..2

Return values:
<MaxLength> Range: 1 to 200
Increment: 1
*RST: 20
Default unit: UI

Usage: Query only

ADVJitter<m>:DCOMposition:RLEStimate

Runs an estimate of the step response for the advanced jitter analysis measurements.

Suffix:
<m> 1..2

Usage: Event
Asynchronous command

ADVJitter<m>:DCOMposition:POSStep <Position>

Sets the steps position.

Suffix:
<m> 1..2

Parameters:
<Position> Range: 0 to 100
Increment: 1
*RST: 50
Default unit: %

ADVJitter<m>:DCOMposition:RESPlength <Length>

Set the step response length in unit intervals.

Suffix:
<m> 1..2

Parameters:
<Length> Range: 1 to 200
Increment: 1
*RST: 20
Default unit: UI

ADVJitter<m>:DCOMposition:ENP <State>

Enables the display of the periodic jitter components table.

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: ON

ADVJitter<m>:DCOMposition:NHORizontal <MaxNumber>

Sets the maximum number of periodic components in the horizontal direction that are considered for the measurement. If the actual number of periodic components is higher than the set value, they are considered for the random jitter component.

Suffix:
<m> 1..2

Parameters:
<MaxNumber> Range: 0 to 10
Increment: 1
*RST: 5

ADVJitter<m>:DCOMposition:NVERtical <MaxNumber>

Sets the maximum number of periodic components in the vertical direction that are considered for the measurement. If the actual number of periodic components is higher than the set value, they are considered for the random jitter component.

Suffix:
<m> 1..2

Parameters:
<MaxNumber> Range: 0 to 10
Increment: 1
*RST: 5

ADVJitter<m>:DCOMposition:ENOBu <State>

Suffix:
<m> 1..2

Parameters:
<State> ON | OFF
*RST: OFF

24.22.3 Clock data recovery (software-based, option R&S RTO6-K12)

CDR:SOFTware<m>:ALGorithm....2510
CDR:SOFTware<m>:SERStandard....2510
CDR:SOFTware<m>:ESBRate:SOURce....2511
CDR:SOFTware<m>:ESBRate:BREStimate....2511
CDR:SOFTware<m>:ESBRate:CONTinuous....2511
CDR:SOFTware<m>:BITRate....2512
CDR:SOFTware<m>:ESLope....2512
CDR:SOFTware<m>:SYNC....2512
CDR:SOFTware<m>:RESults....2513
CDR:SOFTware<m>:PLL:ORDer....2513
CDR:SOFTware<m>:PLL:BWIDth....2514
CDR:SOFTware<m>:PLL:RELBwidth....2514
CDR:SOFTware<m>:PLL:DAMPing....2514
CDR:SOFTware<m>:REVLockpll....2514
CDR:SOFTware<m>:SAMTime....2515

CDR:SOFTware<m>:ALGorithm <Algorithm>

Sets the software algorithm that is used for clock data recovery.

Suffix:

<m> 1..2
Number of the software CDR setup

Parameters:

<Algorithm> CFRequency | PLL | FF | PLLStart

CFRequency
Constant frequency. CDR uses the nominal bit rate to generate the clock signal. The method assumes that the frequency of the signal is constant during the complete acquisition.

PLL
Phase-locked loop control system. It can follow slow deviations in the frequency of the data stream.

FF
Feed forward algorithm, a mathematical optimization method. Feed forward is available if none of the listed serial standards is used ("Custom"). It is faster than PLL, i.e., the settling time is shorter. Feed forward is also more precise than PLL. To set up the algorithm, set the bandwidth and the relative bandwith.

*RST: PLLStart

Usage: Asynchronous command

CDR:SOFTware<m>:SERStandard <SerialStandard>

Sets the serial standard. The nominal bit rate and bandwidth settings are set according to the selected standard, no estimation is required. The serial standards are available only if the instrument is able to support its maximum nominal bit rate.

Set `CUSTOM` to adjust the bit rate and the bandwidth manually.

Suffix:
<m> 1..2
Number of the software CDR setup

Parameters:
<SerialStandard> CUSTOM | USB20 | USB31GEN1 | USB31GEN2 | DISPPORT11 | DISPPORT12 | FIBCHAN1 | FIBCHAN2 | FIBCHAN4 | FIBCHAN8 | FIBCHAN16 | HDMI12 | HDMI14 | HDMI20 | PCIGEN1 | PCIGEN2 | PCIGEN3 | PCIGEN4 | SATAGEN1500 | SATAGEN11667 | SATAGEN1 | SATAGEN2500 | SATAGEN21667 | SATAGEN2 | SATAGEN3500 | SATAGEN31667 | SATAGEN3 | SASGEN1 | SASGEN2 | SASGEN3 | TB10G | XAUI | CUSTOM | USB20 | USB31GEN1 | DISPPORT11 | DISPPORT12 | FIBCHAN1 | FIBCHAN2 | FIBCHAN4 | HDMI12 | HDMI14 | HDMI20 | PCIGEN1 | PCIGEN2 | SATAGEN1500 | SATAGEN11667 | SATAGEN1 | SATAGEN2500 | SATAGEN21667 | SATAGEN2 | SATAGEN3500 | SATAGEN31667 | SATAGEN3 | SASGEN1 | SASGEN2 | XAUI

*RST: CUSTOM

Usage: Asynchronous command

CDR:SOFTware<m>:ESBRate:SOURce <BitrateSource>

Selects the source of the data stream, which is analized to recover the clock.

Suffix:
<m> 1..2
Number of the software CDR setup

Parameters:
<BitrateSource> C1W1 | C2W1 | C3W1 | C4W1 | R1 | R2 | R3 | R4

Usage: Asynchronous command

CDR:SOFTware<m>:ESBRate:BREStimate

Estimates the bit rate once, for the current acquisition.

Suffix:
<m> 1..2
Number of the software CDR setup

Usage: Event
Asynchronous command

CDR:SOFTware<m>:ESBRate:CONTinuous <BitrateEstimation>

Enables ongoing correction of the bit rate.

Suffix:
<m> 1..2
Number of the software CDR setup

Parameters:
<BitrateEstimation> ON | OFF
*RST: OFF

Usage: Asynchronous command

CDR:SOFTware<m>:BITRate <Bitrate>

Sets the quiescent frequency of the PLL. It corresponds to the data rate of the data stream from which the clock is to be recovered.

Suffix:
<m> 1..2
Number of the software CDR setup

Parameters:
<Bitrate> Range: 100 to max {5 Gbps, <available instrument bandwidth}
Increment: 10
*RST: 1E+9
Default unit: bps

Usage: Asynchronous command

CDR:SOFTware<m>:ESLope <Edge>

Selects the edges of the data stream that are used for the clock recovery.

- "Either": Both positive and negative edges are used
- "Positive / Negative": Only one edge direction is used. Use one of these settings if the other edge might deliver unreliable results.

Suffix:
<m> 1..2
Number of the software CDR setup

Parameters:
<Edge> POSitive | NEGative | EITHer
*RST: EITHer

Usage: Asynchronous command

CDR:SOFTware<m>:SYNC <InitialSync>

Defines the phase reference for the first clock edge.

Suffix:
<m> 1..2
Number of the software CDR setup

Parameters:
<InitialSync> SAMPle | DATaedge

SAMPle
The first clock edge matches the first sample of the waveform at the left border of the display.

DATaedge
The first clock edge matches the first edge of the data signal.

*RST: DATaedge

Usage: Asynchronous command

CDR:SOFTware<m>:RESults <Results>

The PLL requires some time to synchronize to the phase of the data stream. You can select when the CDR algorithm returns clock edges.

Suffix:
<m> 1..2
Number of the software CDR setup

Parameters:
<Results> ALL | AISYnc

ALL
All clock edges are used.

AISYnc
The clock edges of the synchronization time are discarded; results are gathered after initial synchronization of the CDR. Thus, meaningful TIE measurement results can be obtained.

*RST: ALL

Usage: Asynchronous command

CDR:SOFTware<m>:PLL:ORDer <PLLOrder>

Sets the order of the PLL: first or second order. PLL of higher order can compensate for more complex jitter behavior.

Suffix:
<m> 1..2
Number of the software CDR setup

Parameters:
<PLLOrder> FIRSt | SECond

*RST: SECond

Usage: Asynchronous command

CDR:SOFTware<m>:PLL:BWIDth <PLLBandwidth>

Sets the PLL bandwidth. It defines the part of the spectrum that the PLL can follow during synchronization. The PLL bandwidth is usually defined by the transmission standard.

Suffix:
<m> 1..2
Number of the software CDR setup

Parameters:
<PLLBandwidth> Range: Nominal BITRate * Range of RELBwidth (dependent range)
Increment: 10
*RST: 599.88E+3
Default unit: Hz

Usage: Asynchronous command

CDR:SOFTware<m>:PLL:RELBwidth <PLLRelBw>

Sets the relative bandwidth, that is the ratio of the nominal bit rate to the PLL bandwidth.

Suffix:
<m> 1..2
Number of the software CDR setup

Parameters:
<PLLRelBw> Range: 10 to 5000
Increment: 1
*RST: 1667

Usage: Asynchronous command

CDR:SOFTware<m>:PLL:DAMPing <DampingFactor>

Sets the damping factor, which is only relevant for second order PLL.

Suffix:
<m> 1..2
Number of the software CDR setup

Parameters:
<DampingFactor> Range: 0.5 to 1
Increment: 0.01
*RST: 0.7

Usage: Asynchronous command

CDR:SOFTware<m>:REVLockpll <ReverseLockPll>

Enables the reverese lock PLL algorithm. The signal is locked on the first edge.

Suffix:
<m> 1..2

Parameters:
<ReverseLockPll> ON | OFF
*RST: ON

Usage: Asynchronous command

CDR:SOFTware<m>:SAMTime <UntIntvlOffs>

Sets a sampling time for the clock signal, an offset for the clock edge in relation to the bit start. The clock edge sets the beginning of the unit interval. The sampling time is a number between 0 and 1. Value 0 sets the clock edge to the beginning of the bit period; value 0.5 sets the clock edge to the middle of the bit period.

Suffix:
<m> 1..2

Parameters:
<UntIntvlOffs> Range: 0 to 1
Increment: 0.01
*RST: 0.5
Default unit: UI

Usage: Asynchronous command

24.22.4 Clock data recovery (hardware-based, option R&S RTO6-K13)

CDR:HARDware:SOURce 2515
CDR:HARDware:SERStandard 2516
CDR:HARDware:BREStimate 2516
CDR:HARDware:BITRate 2516
CDR:HARDware:ESLope 2516
CDR:HARDware:PLL:ORDer 2516
CDR:HARDware:PLL:BWIDth 2517
CDR:HARDware:PLL:RELBwidth 2517
CDR:HARDware:PLL:DAMPing 2517
CDR:HARDware:SAMTime 2517
TRIGger<m>:LEVel<n>[:VALue] 2518

CDR:HARDware:SOURce <AnalogSource>

Selects the channel signal that is used for clock recovery.

Parameters:
<AnalogSource> C1 | C2 | C3 | C4
*RST: C1

Usage: Asynchronous command

CDR:HARDware:SERStandard <SerialStandard>

Sets the serial standard. Set `CUSTOM` to adjust the bit rate and the bandwidth manually.

Parameters:

<SerialStandard>	*RST:	CUSTOM

Usage: Asynchronous command

CDR:HARDware:BREStimate

Estimates the bite rate once, for the current acquisition.

Usage: Event
Asynchronous command

CDR:HARDware:BITRate <Bitrate>

Parameters:

<Bitrate>	Range:	200E+3 to 2.5E+9 for 10 GSa/s; range depends on sample rate
	Increment:	10
	*RST:	1E+9
	Default unit:	bps

Usage: Asynchronous command

CDR:HARDware:ESLope <Edge>

Selects the edges of the data stream that are used for the clock recovery.

- "Either": Both positive and negative edges are used
- "Positive / Negative": Only one edge direction is used. Use one of these settings if the other edge might deliver unreliable results.

Parameters:

<Edge>	POSitive | NEGative | EITHer	
	*RST:	EITHer

Usage: Asynchronous command

CDR:HARDware:PLL:ORDer <PLLOrder>

Sets the order of the PLL: first or second order. PLL of higher order can compensate for more complex jitter behavior.

Parameters:

<PLLOrder>	FIRSt | SECond	
	*RST:	FIRSt

Usage: Asynchronous command

CDR:HARDware:PLL:BWIDth <PLLBandwidth>

Sets the PLL bandwidth. It defines the part of the spectrum that the PLL can follow during synchronization. The PLL bandwidth is usually defined by the transmission standard.

Parameters:

<PLLBandwidth>	Range:	Nominal BITRate * Range of RELBwidth (dependent range)
	Increment:	10
	*RST:	599.88E+3
	Default unit:	Hz

Usage: Asynchronous command

CDR:HARDware:PLL:RELBwidth <PLLRelBw>

Sets the relative bandwidth, that is the ratio of the nominal bit rate to the PLL bandwidth.

Parameters:

<PLLRelBw>	Range:	500 to 3000
	Increment:	1
	*RST:	1667

Usage: Asynchronous command

CDR:HARDware:PLL:DAMPing <DampingFactor>

Sets the damping factor, which is only relevant for second order PLL.

Parameters:

<DampingFactor>	Range:	0.5 to 1
	Increment:	0.01
	*RST:	0.7

Usage: Asynchronous command

CDR:HARDware:SAMTime <UntIntvlOffs>

Sets a sampling time for the clock signal, an offset for the clock edge in relation to the bit start. The clock edge sets the beginning of the unit interval. The sampling time is a number between 0 and 1. Value 0 sets the clock edge to the beginning of the bit period; value 0.5 sets the clock edge to the middle of the bit period.

Parameters:

<UntIntvlOffs>	Range:	0 to 1
	Increment:	0.01
	*RST:	0.5
	Default unit:	UI

Usage: Asynchronous command

TRIGger<m>:LEVel<n>[:VALue] <Level>

Sets the trigger level for the specified event and source.

If the trigger source is serial bus, the trigger level is set by the thresholds in the protocol configuration.

Suffix:

<m> 1..3
1 = A-trigger, 2 = B-trigger, 3 = R-trigger

<n> 1..31
Indicates the trigger source:
1...4 = channel 1 to 4
5 = external trigger input
6...11 = not available
12...27 = R&S RT-ZVC input channels
28...31 = differential signals Diff1, Diff2 and Common1, Common2

Parameters:

<Level> Voltage for the trigger level.

Range: Depends on vertical scale, channel offset and other settings. The trigger level must be within the current display range.
Increment: 1E-3
*RST: 0
Default unit: V

Example: `TRIG:LEV5 0.01`
Sets the trigger level for the external trigger signal to 10 mV.
`TRIG2:LEV3 0.2`
Sets the trigger level for the B-event and B-trigger source channel 3 to 200 mV.

Usage: Asynchronous command

24.22.5 Serial pattern trigger using CDR

TRIGger<m>:SPATtern:CDR 2518
CDR:HARDware:SAMTime 2519

TRIGger<m>:SPATtern:CDR <CdrInUse>

Disables the clock source and enables the recovered clock signal generated by hardware CDR.

Suffix:

<m> 1..3
Only 1 = A-trigger, 2 | 3 = not available. Can be omitted.

Parameters:
<CdrInUse> ON | OFF
*RST: OFF

Usage: Asynchronous command

CDR:HARDware:SAMTime <UntIntvlOffs>

Sets a sampling time for the clock signal, an offset for the clock edge in relation to the bit start. The clock edge sets the beginning of the unit interval. The sampling time is a number between 0 and 1. Value 0 sets the clock edge to the beginning of the bit period; value 0.5 sets the clock edge to the middle of the bit period.

Parameters:
<UntIntvlOffs> Range: 0 to 1
Increment: 0.01
*RST: 0.5
Default unit: UI

Usage: Asynchronous command

24.22.6 Eye mask testing

The eye mask testing is available if the option R&S RTO6-K12 is activated on the instrument.

The commands for eye mask testing are described in Chapter 24.14.6, "Eye mask testing", on page 1599.

24.23 Power analysis (option R&S RTO6-K31)

- General 2520
- Deskew 2520
- Report 2522
- Power quality 2527
- Inrush current 2529
- Current harmonic 2531
- Modulation analysis 2533
- Dynamic ON resistance 2535
- Slew rate 2536
- S.O.A. 2539
- Turn on/off 2541
- Switching loss 2543
- Power efficiency 2546
- Ripple 2547
- Transient response 2554
- Spectrum 2557

24.23.1 General

POWer:ENABle 2520
POWer:SOURce:CURRent<1..2> 2520
POWer:SOURce:VOLTage<1..4> 2520

POWer:ENABle

Activates the power mode and initializes the power measurements. If the power mode is disabled, the instrument does not accept any `POWer` command.

Use `POWer:ENABle` after each `*RST`.

Example: See Chapter 24.3.8.1, "Auto deskew", on page 1277

Usage: Event
Asynchronous command

POWer:SOURce:CURRent<1..2> <CurrentSource>

Sets the channel for the current source.

Parameters:

<CurrentSource> CHAN1 | CHANnel1 | CHAN2 | CHANnel2 | CHAN3 | CHANnel3 | CHAN4 | CHANnel4

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4

*RST: CURRent1: CHAN2, CURRent2: CHAN4,

Usage: Asynchronous command

POWer:SOURce:VOLTage<1..4> <VoltageSource>

Sets the channel for the voltage source input.

Parameters:

<VoltageSource> CHAN1 | CHANnel1 | CHAN2 | CHANnel2 | CHAN3 | CHANnel3 | CHAN4 | CHANnel4

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4

*RST: VOLTage1: CHAN1, VOLTage2: CHAN3, VOLTage3: CHAN3, VOLTage4: CHAN4

Usage: Asynchronous command

24.23.2 Deskew

Programming example: Chapter 24.3.8.1, "Auto deskew", on page 1277

POWer:DESKew:CURRent....2521
POWer:DESKew:EXECute....2521
POWer:DESKew:RESet....2521
POWer:DESKew:TIME?....2521
POWer:DESKew:UDPReset....2521

POWer:DESKew:CURRent

Applies the result of the auto deskew to the "Skew offset" value.

Usage: Event

POWer:DESKew:EXECute

Starts the auto deskew.

Usage: Event

POWer:DESKew:RESet <OverwriteCurrSkew>

Overwrites the present skew setup.

Parameters:
<OverwriteCurrSkew> ON | OFF
*RST: ON

POWer:DESKew:TIME?

Queries the result of the auto deskew.

Return values:
<AutoDeskewOffs> Range: -100E-9 to 100E-9
*RST: 0
Default unit: s

Usage: Query only

POWer:DESKew:UDPReset <UsrDefinedPreset>

Activates or deactivates a user defined setup. If ON, the instrument setup including probe setup and the deskew values are written to a user defined preset file (saveset) that can be loaded using `MMEMory:RCL` on page 1646.

The default path is:

`C:\Users\Public\Documents\Rohde-Schwarz\RTO6\SaveSets\`

Parameters:
<UsrDefinedPreset> ON | OFF
*RST: ON

24.23.3 Report

POWer:REPort:CONTent:HSETup....2522
POWer:REPort:CONTent:MSETup....2522
POWer:REPort:CONTent:MSIGnal....2522
POWer:REPort:CONTent:RESU....2522
POWer:REPort:CONTent:SETTings....2522
POWer:REPort:CONTent:TSETup....2522
POWer:REPort:CONTent:VSETup....2522
POWer:REPort:CONTent:TITLe....2523
POWer:REPort:DESCription....2523
POWer:REPort:DUT....2523
POWer:REPort:SITE....2523
POWer:REPort:TEMPerature....2523
POWer:REPort:USER....2523
POWer:REPort:FONT:COLO....2523
POWer:REPort:FONT:FAMI....2523
POWer:REPort:FONT:SIZE....2523
POWer:REPort:LOGO....2523
POWer:REPort:PAPersize....2524
POWer:REPort:FILE:NAME....2524
POWer:REPort:FILE:DELete....2524
POWer:REPort:FILE:NEW....2524
POWer:REPort:FILE:SAVE....2524
POWer:REPort:TEST:ADD....2524
POWer:REPort:TEST:INSert....2524
POWer:REPort:TEST:REMove....2524
POWer:REPort:INVert....2525
POWer:REPort:TEST:DSEA....2525
POWer:REPort:TEST:ISE....2525
POWer:REPort:TEST:SEA....2525
POWer:REPort:TEST:RSE....2525
POWer:REPort:TEST:DIRectory....2526
POWer:REPort:TEST:COMMent....2526
POWer:REPort:TEST:COUNt....2526
POWer:REPort:TEST:LSENd?....2527

POWer:REPort:CONTent:HSETup <ContentHorizSet>
POWer:REPort:CONTent:MSETup <ContentMeasSet>
POWer:REPort:CONTent:MSIGnal <ContentMeasuredSigns>
POWer:REPort:CONTent:RESU <ContentResults>
POWer:REPort:CONTent:SETTings <ContentSettings>
POWer:REPort:CONTent:TSETup <ContentTrigSet>
POWer:REPort:CONTent:VSETup <ContentVertSet>

Sets how often the respective content is shown in the final report.

Parameters:

<ContentVertSet> ALWAYS | NEVER | ONCE

*RST: ONCE

POWer:REPort:CONTent:TITLe <ContentTitle>

Includes the title page in the report.

Parameters:

<ContentTitle> ON | OFF

*RST: ON

POWer:REPort:DESCription <String>
POWer:REPort:DUT <String>
POWer:REPort:SITE <String>
POWer:REPort:TEMPerature <String>
POWer:REPort:USER <String>

The content of the strings is shown at the title page of a report if the title page is included in the report.

Parameters:

<String>

POWer:REPort:FONT:COLO <FontColor>

Sets the font color.

Parameters:

<FontColor> Range: 0 to 4294967295
Increment: 1
*RST: 0

POWer:REPort:FONT:FAMI <FontFamily>

Selects the font family. You can choose between Arial and Helvetica.

Parameters:

<FontFamily> ARIAL | HELV

*RST: ARIAL

POWer:REPort:FONT:SIZE <FontSize>

Sets the font size.

Parameters:

<FontSize> Range: 10 to 30
Increment: 1
*RST: 12

POWer:REPort:LOGO <LogoFile>

Selects a path to a logo picture file.

Parameters:
<LogoFile>

POWer:REPort:PAPersize <PaperSize>

Set the layout of your report.

Parameters:
<PaperSize> A4 | USL
*RST: A4

POWer:REPort:FILE:NAME <Path>

Defines the path and file name of the report file that is to be created, saved, or deleted.

Parameters:
<Path> String containing path and file name

POWer:REPort:FILE:DELete

Deletes the selected report file.

Usage: Event

POWer:REPort:FILE:NEW

Creates a new report file.

Usage: Event

POWer:REPort:FILE:SAVE

Saves the report file.

Usage: Event

POWer:REPort:TEST:ADD <MeasType>
POWer:REPort:TEST:INSert <MeasType>, <Index>
POWer:REPort:TEST:REMove <MeasType>, <Index>

Manage reports.

Setting parameters:
<MeasType> QUAL | RUSH | HARM | MODU | DONR | SLEW | SOA | TURN | SWIT | EFF | RIPP | TRANS | SPEC

QUAL
Power Quality

RUSH
Inrush Current

HARM
Current Harmonic

MODU
Modulation Analysis

DONR
Dynamic ON Resistance

SLEW
Slew Rate

SOA
Safe Operating Area (S.O.A.)

TURN
Turn On/Off

SWIT
Switching Loss

EFF
Power Efficiency

RIPP
Ripple

TRANS
Transient Response

SPEC
Spectrum

<Index>

Usage: Setting only

POWer:REPort:INVert <InvertScreenshotClr>
POWer:REPort:TEST:DSEA <MeasType>
POWer:REPort:TEST:ISE <MeasType>
POWer:REPort:TEST:SEA <MeasType>
POWer:REPort:TEST:RSE <MeasType>

Manage the selection of reports.

Parameters:

<MeasType> QUAL | RUSH | HARM | MODU | DONR | SLEW | SOA | TURN | SWIT | EFF | RIPP | TRANS | SPEC

QUAL
Power Quality

RUSH
Inrush Current

HARM
Current Harmonic

MODU
Modulation Analysis

DONR
Dynamic ON Resistance

SLEW
Slew Rate

SOA
Safe Operating Area (S.O.A.)

TURN
Turn On/Off

SWIT
Switching Loss

EFF
Power Efficiency

RIPP
Ripple

TRANS
Transient Response

SPEC
Spectrum

POWer:REPort:TEST:DIRectory <MeasType>, <DirectoryPath>
POWer:REPort:TEST:DIRectory? <MeasType>

Selects the directory, in which the reports are saved.

Setting parameters:
<DirectoryPath>

Parameters for setting and query:
<MeasType> QUAL | RUSH | HARM | MODU | DONR | SLEW | SOA | TURN | SWIT | EFF | RIPP | TRANS | SPEC

POWer:REPort:TEST:COMMent <MeasType>, <Comment>
POWer:REPort:TEST:COMMent? <MeasType>

Sets a comment for the report.

Setting parameters:
<Comment>

Parameters for setting and query:
<MeasType> QUAL | RUSH | HARM | MODU | DONR | SLEW | SOA | TURN | SWIT | EFF | RIPP | TRANS | SPEC

POWer:REPort:TEST:COUNt <MeasType>

Parameters:
<MeasType> QUAL | RUSH | HARM | MODU | DONR | SLEW | SOA | TURN | SWIT | EFF | RIPP | TRANS | SPEC

Return values:
<Count>

POWer:REPort:TEST:LSENd? <MeasType>

Query parameters:
<MeasType> QUAL | RUSH | HARM | MODU | DONR | SLEW | SOA | TURN | SWIT | EFF | RIPP | TRANS | SPEC

Usage: Query only

24.23.4 Power quality

POWer:QUALity:AUTO....2527
POWer:QUALity:EXECute....2527
POWer:QUALity:FREQ....2527
POWer:QUALity:FCUS....2528
POWer:QUALity:REPort:ADD....2528
POWer:QUALity:RESult:CURRent:CREStfactor?....2528
POWer:QUALity:RESult:CURRent:FREQuency?....2528
POWer:QUALity:RESult:CURRent:PEAK?....2528
POWer:QUALity:RESult:CURRent:RMS?....2528
POWer:QUALity:RESult:POWer:APParent?....2528
POWer:QUALity:RESult:POWer:PFACtor?....2528
POWer:QUALity:RESult:POWer:PHASe?....2528
POWer:QUALity:RESult:POWer:REACtive?....2528
POWer:QUALity:RESult:POWer:REALpower?....2528
POWer:QUALity:RESult:VOLTage:CREStfactor?....2528
POWer:QUALity:RESult:VOLTage:FREQuency?....2528
POWer:QUALity:RESult:VOLTage:PEAK?....2528
POWer:QUALity:RESult:VOLTage:RMS?....2528

POWer:QUALity:AUTO <Autoscale>

Selects the scaling for the display of the results.

Parameters:
<Autoscale> AUTO | MANUAL
*RST: MANUAL

POWer:QUALity:EXECute

Starts the power quality measurement.

Usage: Event

POWer:QUALity:FREQ <Frequency>

Sets the input frequency of the source signal in Hz.

Parameters:

<Frequency> F50 | F60 | F360 | F400 | F650 | F800 | NFF650 | WFF800 | FCUS

NFF650: 360 to 650 Hz
WFF800: 360 to 800 Hz
FCUS: user-defined frequency to be set using `POWer:QUALity:FCUS`.

*RST: F50

POWer:QUALity:FCUS <CustomFrequency>

Sets the user-defined frequency if `POWer:QUALity:FREQ` is set to `FCUS`.

Parameters:

<CustomFrequency> Range: 1 to 5000
Increment: 1
*RST: 16.666
Default unit: Hz

Firmware/software: Version 2.70

POWer:QUALity:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:QUALity:RESult:CURRent:CREStfactor?
POWer:QUALity:RESult:CURRent:FREQuency?
POWer:QUALity:RESult:CURRent:PEAK?
POWer:QUALity:RESult:CURRent:RMS?
POWer:QUALity:RESult:POWer:APParent?
POWer:QUALity:RESult:POWer:PFACtor?
POWer:QUALity:RESult:POWer:PHASe?
POWer:QUALity:RESult:POWer:REACtive?
POWer:QUALity:RESult:POWer:REALpower?
POWer:QUALity:RESult:VOLTage:CREStfactor?
POWer:QUALity:RESult:VOLTage:FREQuency?
POWer:QUALity:RESult:VOLTage:PEAK?
POWer:QUALity:RESult:VOLTage:RMS?

Returns the value of the respective result.

Return values:

<Value> Range: Depends on the measured quantity.
Default unit: Depends on the measured quantity.

Usage: Query only

24.23.5 Inrush current

This measurement is a single shot measurement. To start the measurement, use the `RUNS` command.

POWer:INRush:ADD 2529
POWer:INRush:INSert 2529
POWer:INRush:REMove 2529
POWer:INRush:COUNt? 2529
POWer:INRush:EXECute 2530
POWer:INRush:GATE<m>:STARt 2530
POWer:INRush:GATE<m>:STOP 2530
POWer:INRush:GATE<m>:VALue 2530
POWer:INRush:MAXCurrent 2530
POWer:INRush:TRIGger 2530
POWer:INRush:REPort:ADD 2531

POWer:INRush:ADD

Adds a gate.

Usage: Event

POWer:INRush:INSert <GateIndex>

Inserts a gate.

Setting parameters:
<GateIndex>

Usage: Setting only

POWer:INRush:REMove <GateIndex>

Removes a gate

Setting parameters:
<GateIndex>

Usage: Setting only

POWer:INRush:COUNt?

Queiries the number of inrush current gates.

Return values:
<Count>

Usage: Query only

POWer:INRush:EXECute

Starts the inrush current measurement.

Usage: Event

POWer:INRush:GATE<m>:STARt <StartTime>
POWer:INRush:GATE<m>:STOP <StopTime>

Sets the measuring time for the selected gate.

Suffix:
<m> *

Parameters:
<StopTime>
Range: 0 to 10
Increment: 0
*RST: 100E-6
Default unit: s

POWer:INRush:GATE<m>:VALue <Value>

Returns the value of the inrush current.

Suffix:
<m> *

Parameters:
<Value>
Range: -1000 to 1000
Increment: 0
*RST: 0
Default unit: A

POWer:INRush:MAXCurrent <MaxExpCurr>

Sets the maximum expected current for the vertical scale.

Parameters:
<MaxExpCurr>
Range: -1000 to 1000
Increment: 0
*RST: 10
Default unit: A

POWer:INRush:TRIGger <CurrentValue>

Sets the current value for the trigger.

Parameters:
<CurrentValue>
Range: -1000 to 1000
Increment: 0
*RST: 1
Default unit: A

POWer:INRush:REPort:ADD

Adds the result to the report list.

Usage: Event

24.23.6 Current harmonic

POWer:HARMonics:AUTO....2531
POWer:HARMonics:DOFR....2531
POWer:HARMonics:ENFR....2531
POWer:HARMonics:MIFR....2532
POWer:HARMonics:EVAL....2532
POWer:HARMonics:EXECute....2532
POWer:HARMonics:REPort:ADD....2532
POWer:HARMonics:RESult<m>:STDinuse?....2532
POWer:HARMonics:RESult<m>:FREQuency<n>:VALue?....2532
POWer:HARMonics:RESult<m>:MAXValue<n>:VALue?....2532
POWer:HARMonics:RESult<m>:STDValue<n>:VALue?....2532
POWer:HARMonics:RESult<m>:VALue<n>:VALue?....2532
POWer:HARMonics:STAN....2533
POWer:HARMonics:VOLT....2533

POWer:HARMonics:AUTO <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale> AUTO | MANUAL

*RST: MANUAL

POWer:HARMonics:DOFR <FrequencyDOA160>

Selects the frequency of the input signal.

Parameters:

<FrequencyDOA160> F360 | F400 | F650 | F800

*RST: F400

POWer:HARMonics:ENFR <FreqEN61000>

Selects the frequency of the input signal.

Parameters:

<FreqEN61000> F50 | F60

*RST: F50

POWer:HARMonics:MIFR <FreqMIL1399>

Selects the frequency of the input signal.

Parameters:
<FreqMIL1399> F400 | F60
*RST: F400

POWer:HARMonics:EVAL <AnalysisRevised>

Sets the evaluation of the results for "Standard" > "RTCA DO-160".

Parameters:
<AnalysisRevised> REVISED | NOREVISED
*RST: NOREVISED

POWer:HARMonics:EXECute

Starts the current harmonic measurement.

Usage: Event

POWer:HARMonics:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:HARMonics:RESult<m>:STDinuse?

Returns the used standard.

Suffix:
<m> 1..2

Return values:
<StandardInUse> ENA | ENB | ENC | END | MIL | RTC
*RST: ENA

Usage: Query only

POWer:HARMonics:RESult<m>:FREQuency<n>:VALue?
POWer:HARMonics:RESult<m>:MAXValue<n>:VALue?
POWer:HARMonics:RESult<m>:STDValue<n>:VALue?
POWer:HARMonics:RESult<m>:VALue<n>:VALue?

Returns the value of the respective result.

Suffix:
<m> 1..2

<n> *

Return values:

<Value> Range: -1000 to 1000
*RST: 0
Default unit: A

Usage: Query only

POWer:HARMonics:STAN <StandardInUse>

Sets a standard for the current harmonic measurement.

Parameters:

<StandardInUse> ENA | ENB | ENC | END | MIL | RTC
*RST: ENA

POWer:HARMonics:VOLT <DOADisplayResult>

Selects if the voltage results are displayed or not for "Standard" > "RTCA DO-160" and enabled "Evaluation with voltage source and revised current law".

Parameters:

<DOADisplayResult> VOLTDISP | NOVOLTDISP
*RST: NOVOLTDISP

Example:

```
POW:HARM:STAN RTC
POW:HARM:EVAL REVISED
POW:HARM:VOLT NOVOLTDISP
```

Selects an evaluation with the revised current law and no voltage display.

24.23.7 Modulation analysis

This measurement is a single shot measurement. To start the measurement, use the `RUNS` command.

POWer:MODulation:AUTO....2534
POWer:MODulation:DHIStogram....2534
POWer:MODulation:EXECute....2534
POWer:MODulation:REPort:ADD....2534
POWer:MODulation:RESult:ACTual?....2534
POWer:MODulation:RESult:AVG?....2534
POWer:MODulation:RESult:EVTCount?....2534
POWer:MODulation:RESult:NPEak?....2534
POWer:MODulation:RESult:PPEak?....2534
POWer:MODulation:RESult:RMS?....2534
POWer:MODulation:RESult:STDDev?....2534

POWer:MODulation:RESult:WFMCount?....2534
POWer:MODulation:SOURce....2535
POWer:MODulation:TYPE....2535

POWer:MODulation:AUTO <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale> AUTO | MANUAL

*RST: MANUAL

POWer:MODulation:DHIStogram <DispHistg>

Activates or deactivates the display of a histogram.

Parameters:

<DispHistg> ON | OFF

*RST: ON

POWer:MODulation:EXECute

Starts the modulation analysis measurement.

Usage: Event
Asynchronous command

POWer:MODulation:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:MODulation:RESult:ACTual? <MeasType>
POWer:MODulation:RESult:AVG? <MeasType>
POWer:MODulation:RESult:EVTCount? <MeasType>
POWer:MODulation:RESult:NPEak? <MeasType>
POWer:MODulation:RESult:PPEak? <MeasType>
POWer:MODulation:RESult:RMS? <MeasType>
POWer:MODulation:RESult:STDDev? <MeasType>
POWer:MODulation:RESult:WFMCount? <MeasType>

Return the specified statistic result of the specified measurement type.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results

- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<MeasType> FREQ | DUTY

Usage: Query only

POWer:MODulation:SOURce <Source>

Selects the source for the measurement.

Parameters:
<Source> CURRENT | VOLTAGE
*RST: VOLTAGE

POWer:MODulation:TYPE <AnalysisType>

Sets the type of measurement.

Parameters:
<AnalysisType> TURNON | CONT
*RST: CONT

24.23.8 Dynamic ON resistance

POWer:DONRes:AUTO....2535
POWer:DONRes:AVG....2535
POWer:DONRes:EXECute....2536
POWer:DONRes:GATE<m>:STARt....2536
POWer:DONRes:GATE<m>:STOP....2536
POWer:DONRes:REPort:ADD....2536
POWer:DONRes:RESult:RESistance?....2536

POWer:DONRes:AUTO <Autoscale>

Selects the scaling for the display of the results.

Parameters:
<Autoscale> AUTO | MANUAL
*RST: MANUAL

POWer:DONRes:AVG <Average>

Enables/disables averaging.

Parameters:
<Average> ON | OFF
*RST: ON

POWer:DONRes:EXECute

Starts the dynamic on resistance measurement.

Usage: Event
Asynchronous command

POWer:DONRes:GATE<m>:STARt <Start>
POWer:DONRes:GATE<m>:STOP <Stop>

Sets the value for the cursor.

Suffix:
<m> 1..2

Parameters:
<Stop>

POWer:DONRes:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:DONRes:RESult:RESistance?

Returns the the dynamic on resistance value.

Return values:
<Resistance> Range: -100E+24 to 100E+24
*RST: 0
Default unit: Ω

Usage: Query only

24.23.9 Slew rate

POWer:SLEWrate:SOURce 2537
POWer:SLEWrate:AUTO 2537
POWer:SLEWrate:AVGDeriv 2537
POWer:SLEWrate:EXECute 2537
ACQuire:ARESet:MODE 2537
ACQuire:ARESet:TIME 2538
ACQuire:ARESet:COUNt 2538
POWer:SLEWrate:GATE:STARt 2538
POWer:SLEWrate:GATE:STOP 2538
POWer:SLEWrate:REPort:ADD 2538
POWer:SLEWrate:RESult:ACTual? 2538
POWer:SLEWrate:RESult:AVG? 2538
POWer:SLEWrate:RESult:EVTCount? 2538
POWer:SLEWrate:RESult:NPEak? 2538

POWer:SLEWrate:RESult:PPEak?....2538
POWer:SLEWrate:RESult:RMS?....2538
POWer:SLEWrate:RESult:STDDev?....2539
POWer:SLEWrate:RESult:WFMCount?....2539

POWer:SLEWrate:SOURce <Source>

Selects the source for the slew rate measurement.

Parameters:

<Source> CURRENT | VOLTAGE

*RST: VOLTAGE

POWer:SLEWrate:AUTO <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale> AUTO | MANUAL

*RST: MANUAL

POWer:SLEWrate:AVGDeriv <AvgDerivative>

Activates or deactivates average.

Parameters:

<AvgDerivative> ON | OFF

*RST: ON

POWer:SLEWrate:EXECute

Starts the slew rate measurement.

Usage: Event

ACQuire:ARESet:MODE <ArtmRst>

Defines when the envelope and average evaluation restarts.

Parameters:

<ArtmRst> NONE | TIME | WFMS

TIME
Restarts the envelope and average calculation after the time defined with `ACQuire:ARESet:TIME`.

WFMS
Restarts the envelope and average calculation after a number of acquired waveforms defined with `ACQuire:ARESet:COUNt` on page 1335.

*RST: NONE

Usage: Asynchronous command

ACQuire:ARESet:TIME <EnvelopeTimeout>

Defines the time after which the envelope and average evaluation restarts.

The setting is relevant if `ACQuire:ARESet:MODE` is set to `TIME`.

Parameters:

<EnvelopeTimeout>
Range: 0.1 to 10000
Increment: 0.01
*RST: 0.1
Default unit: s

Usage: Asynchronous command

ACQuire:ARESet:COUNt <NofWaveforms>

Defines the number of acquired waveforms after which the envelope and average evaluation restarts.

The setting is relevant if `ACQuire:ARESet:MODE` is set to `WFMS`.

Parameters:

<NofWaveforms>
Range: 2 to 16777215
Increment: 10
*RST: 1000

Usage: Asynchronous command

POWer:SLEWrate:GATE:STARt <T0>
POWer:SLEWrate:GATE:STOP <T1>

Sets the value for the cursor.

Parameters:

<T1>

POWer:SLEWrate:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:SLEWrate:RESult:ACTual? <MeasType>
POWer:SLEWrate:RESult:AVG? <MeasType>
POWer:SLEWrate:RESult:EVTCount? <MeasType>
POWer:SLEWrate:RESult:NPEak? <MeasType>
POWer:SLEWrate:RESult:PPEak? <MeasType>
POWer:SLEWrate:RESult:RMS? <MeasType>

POWer:SLEWrate:RESult:STDDev? <MeasType>
POWer:SLEWrate:RESult:WFMCount? <MeasType>

Return the specified statistic result of the specified measurement type.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<MeasType> MIN | MAX

Usage: Query only

24.23.10 S.O.A

POWer:SOA:EXECute....2539
POWer:SOA:LINear:ADD....2539
POWer:SOA:LOGarithmic:ADD....2539
POWer:SOA:LINear:COUNt?....2540
POWer:SOA:LOGarithmic:COUNt?....2540
POWer:SOA:LINear:REMove....2540
POWer:SOA:LOGarithmic:REMove....2540
POWer:SOA:LINear:INSert....2540
POWer:SOA:LOGarithmic:INSert....2540
POWer:SOA:LINear:POINt<m>:CURRent....2540
POWer:SOA:LOGarithmic:POINt<m>:CURRent....2540
POWer:SOA:LINear:POINt<m>:VOLTage....2540
POWer:SOA:LOGarithmic:POINt<m>:VOLTage....2540
POWer:SOA:MASK....2541
POWer:SOA:REPort:ADD....2541
POWer:SOA:SCALe....2541
POWer:SOA:SWITch....2541

POWer:SOA:EXECute

Starts the safe operating area measurement.

Usage: Event

POWer:SOA:LINear:ADD
POWer:SOA:LOGarithmic:ADD

Adds a point.

Usage: Event

POWer:SOA:LINear:COUNt?
POWer:SOA:LOGarithmic:COUNt?

Queries the number of points.

Return values:
<Count>

Usage: Query only

POWer:SOA:LINear:REMove <GateIndex>
POWer:SOA:LOGarithmic:REMove <GateIndex>

Removes a point.

Setting parameters:
<GateIndex>

Usage: Setting only

POWer:SOA:LINear:INSert <GateIndex>
POWer:SOA:LOGarithmic:INSert <GateIndex>

Inserts a point.

Setting parameters:
<GateIndex>

Usage: Setting only

POWer:SOA:LINear:POINt<m>:CURRent <Amp>
POWer:SOA:LOGarithmic:POINt<m>:CURRent <Amp>

Sets the current value for the respective point.

Suffix:
<m> *

Parameters:
<Amp>
Range: 0.01 to 1000
Increment: 0
*RST: 0.01
Default unit: A

POWer:SOA:LINear:POINt<m>:VOLTage <Volt>
POWer:SOA:LOGarithmic:POINt<m>:VOLTage <Volt>

Sets the voltage value for the respective point.

Suffix:
<m> *

Parameters:

<Volt>	Range:	1E-3 to 1000
	Increment:	0
	*RST:	1E-3
	Default unit:	V

POWer:SOA:MASK <EnableMaskTest>

Activates or deactivates a mask.

Parameters:

<EnableMaskTest>	ON | OFF	
	*RST:	OFF

POWer:SOA:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:SOA:SCALe <Scale>

Sets the scale for the measurement.

Parameters:

<Scale>	LOG | LINEAR	
	*RST:	LOG

POWer:SOA:SWITch <Switch>

Switches between linear and logarithmic scale.

Parameters:

<Switch>	LOGLINEAR | LINEARLOG	
	*RST:	LOGLINEAR

24.23.11 Turn on/off

This measurement is a single shot measurement. To start the measurement, use the `RUNS` command.

POWer:ONOFf:ATOFf 2542
POWer:ONOFf:ATON 2542
POWer:ONOFf:DTOFf 2542
POWer:ONOFf:DTON 2542
POWer:ONOFf:DSOFf 2542
POWer:ONOFf:DSON 2542
POWer:ONOFf:EXECute 2542
POWer:ONOFf:INPut 2542

POWer:ONOFf:REPort:ADD 2542
POWer:ONOFf:RESult:TOFF? 2543
POWer:ONOFf:RESult:TON? 2543
POWer:ONOFf:TIME 2543
POWer:ONOFf:TYPE 2543

POWer:ONOFf:ATOFf <ACTrigLevOff>
POWer:ONOFf:ATON <ACTrigLevOn>

Triggers the beginning of the measurements at the moment the AC input voltage reaches the set value.

Parameters:

<ACTrigLevOn>	Range:	-1E+6 to 1E+6
	Increment:	1E-3
	*RST:	10
	Default unit:	V

POWer:ONOFf:DTOFf <ACTrigLevOff>
POWer:ONOFf:DTON <ACTrigLevOn>
POWer:ONOFf:DSOFf <StateLevelOff>
POWer:ONOFf:DSON <StateLevelOn>

Sets the percentage of the steady state level of the DC output that has to be reached.

Parameters:

<StateLevelOn>	Range:	0 to 100
	Increment:	1
	*RST:	90
	Default unit:	%

POWer:ONOFf:EXECute

Starts the turn on/off measurement.

Usage: Event

POWer:ONOFf:INPut <InputType>

Sets the input type.

Parameters:

<InputType>	AC | DC	
	*RST:	AC

POWer:ONOFf:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:ONOFf:RESult:TOFF?
POWer:ONOFf:RESult:TON?

Returns the result time.

Return values:

<TurnOnTime>
Range: -100E+24 to 100E+24
Increment: 100E-12
*RST: 0
Default unit: s

Usage: Query only

POWer:ONOFf:TIME <Time>

Sets the time, the start of the measurement of the turn off time is delay with, after the trigger point.

Parameters:

<Time>
Range: 100E-12 to 10000
Increment: 100E-9
*RST: 0.1
Default unit: s

POWer:ONOFf:TYPE <MeasType>

Selects the measurement type.

Parameters:

<MeasType>
TON | TOFF

TON - "Turn on" measurement
TOFF - "Turn off" measurement

*RST: TON

24.23.12 Switching loss

POWer:SWITching:AUTO 2544
POWer:SWITching:EXECute 2544
POWer:SWITching:REPort:ADD 2544
POWer:SWITching:SWIFrequency 2544
POWer:SWITching:SWIT 2545
POWer:SWITching:COND 2545
POWer:SWITching:NCON 2545
POWer:SWITching:TON 2545
POWer:SWITching:TOFF 2545
POWer:SWITching:TOTal 2545
POWer:SWITching:GATE:COND:STARt 2545
POWer:SWITching:GATE:COND:STOP 2545
POWer:SWITching:GATE:NCON:STARt 2545
POWer:SWITching:GATE:TOFF:STARt 2545

POWer:SWITching:GATE:TOFF:STOP....2545
POWer:SWITching:GATE:TON:STARt....2545
POWer:SWITching:GATE:TON:STOP....2545
POWer:SWITching:RESult:ENERgy:ACTual?....2545
POWer:SWITching:RESult:ENERgy:AVG?....2545
POWer:SWITching:RESult:ENERgy:EVTCount?....2545
POWer:SWITching:RESult:ENERgy:NPEak?....2545
POWer:SWITching:RESult:ENERgy:PPEak?....2545
POWer:SWITching:RESult:ENERgy:RMS?....2545
POWer:SWITching:RESult:ENERgy:STDDev?....2545
POWer:SWITching:RESult:ENERgy:WFMCount?....2545
POWer:SWITching:RESult:POWer:ACTual?....2546
POWer:SWITching:RESult:POWer:AVG?....2546
POWer:SWITching:RESult:POWer:EVTCount?....2546
POWer:SWITching:RESult:POWer:NPEak?....2546
POWer:SWITching:RESult:POWer:PPEak?....2546
POWer:SWITching:RESult:POWer:RMS?....2546
POWer:SWITching:RESult:POWer:STDDev?....2546
POWer:SWITching:RESult:POWer:WFMCount?....2546

POWer:SWITching:AUTO <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale> AUTO | MANUAL

*RST: MANUAL

POWer:SWITching:EXECute

Starts the switching loss measurement.

Usage: Event
Asynchronous command

POWer:SWITching:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:SWITching:SWIFrequency <SwitchingFreq>

Sets the switching frequency.

Parameters:

<SwitchingFreq> Range: 1 to 500E+9
Increment: 1000
*RST: 10E+6
Default unit: Hz

POWer:SWITching:SWIT <MeasSwitchingFreq>

Activates or deactivates the measurements of the switching frequency.

Parameters:
<MeasSwitchingFreq>ON | OFF
*RST: ON

POWer:SWITching:COND <MeasureConduction>
POWer:SWITching:NCON <MeasureNonConduction>
POWer:SWITching:TON <MeasureTurnOn>
POWer:SWITching:TOFF <MeasureTurnOff>
POWer:SWITching:TOTal <MeasureTotal>

Enables the measurement during the respective period.

Parameters:
<MeasureTotal> ON | OFF
*RST: ON

POWer:SWITching:GATE:COND:STARt <T1>
POWer:SWITching:GATE:COND:STOP <T2>
POWer:SWITching:GATE:NCON:STARt <T3>
POWer:SWITching:GATE:TOFF:STARt <T2>
POWer:SWITching:GATE:TOFF:STOP <T3>
POWer:SWITching:GATE:TON:STARt <T0>
POWer:SWITching:GATE:TON:STOP <T1>

Sets the value for the respective cursor.

Parameters:
<T1>

POWer:SWITching:RESult:ENERgy:ACTual? <MeasType>
POWer:SWITching:RESult:ENERgy:AVG? <MeasType>
POWer:SWITching:RESult:ENERgy:EVTCount? <MeasType>
POWer:SWITching:RESult:ENERgy:NPEak? <MeasType>
POWer:SWITching:RESult:ENERgy:PPEak? <MeasType>
POWer:SWITching:RESult:ENERgy:RMS? <MeasType>
POWer:SWITching:RESult:ENERgy:STDDev? <MeasType>
POWer:SWITching:RESult:ENERgy:WFMCount? <MeasType>

Return the specified statistic result of the specified measurement type.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results

- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<MeasType> SWF | TON | TOF | CON | NCO | TOT

Usage: Query only

POWer:SWITching:RESult:POWer:ACTual? <MeasType>
POWer:SWITching:RESult:POWer:AVG? <MeasType>
POWer:SWITching:RESult:POWer:EVTCount? <MeasType>
POWer:SWITching:RESult:POWer:NPEak? <MeasType>
POWer:SWITching:RESult:POWer:PPEak? <MeasType>
POWer:SWITching:RESult:POWer:RMS? <MeasType>
POWer:SWITching:RESult:POWer:STDDev? <MeasType>
POWer:SWITching:RESult:POWer:WFMCount? <MeasType>

Return the specified statistic result of the specified measurement type.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<MeasType> SWF | TON | TOF | CON | NCO | TOT

Usage: Query only

24.23.13 Power efficiency

POWer:EFFiciency:AUTO....2546
POWer:EFFiciency:EXECute....2547
POWer:EFFiciency:REPort:ADD....2547
POWer:EFFiciency:RESult<m>:ACTual?....2547
POWer:EFFiciency:RESult<m>:AVG?....2547
POWer:EFFiciency:RESult<m>:EVTCount?....2547
POWer:EFFiciency:RESult<m>:NPEak?....2547
POWer:EFFiciency:RESult<m>:PPEak?....2547
POWer:EFFiciency:RESult<m>:RMS?....2547
POWer:EFFiciency:RESult<m>:STDDev?....2547
POWer:EFFiciency:RESult<m>:WFMCount?....2547

POWer:EFFiciency:AUTO <Autoscale>

Selects the scaling for the display of the results.

Parameters:
<Autoscale> AUTO | MANUAL
*RST: MANUAL

POWer:EFFiciency:EXECute

Starts the power efficiency measurement.

Usage: Event

POWer:EFFiciency:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:EFFiciency:RESult<m>:ACTual?
POWer:EFFiciency:RESult<m>:AVG?
POWer:EFFiciency:RESult<m>:EVTCount?
POWer:EFFiciency:RESult<m>:NPEak?
POWer:EFFiciency:RESult<m>:PPEak?
POWer:EFFiciency:RESult<m>:RMS?
POWer:EFFiciency:RESult<m>:STDDev?
POWer:EFFiciency:RESult<m>:WFMCount?

Return the specified statistic result of the specified measurement type.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Suffix:
<m> 1..3

Return values:
<WaveformsCount> Range: 0 to 4E+9
*RST: 0

Usage: Query only

24.23.14 Ripple

POWer:RIPPle:AUToscale 2549
POWer:RIPPle:CURRent 2549
POWer:RIPPle:EXECute 2549

POWer:RIPPle:FREQuency 2549
POWer:RIPPle:REPort:ADD 2550
POWer:RIPPle:RESult:FREQuency[:ACTual]? 2550
POWer:RIPPle:RESult:FREQuency:AVG? 2550
POWer:RIPPle:RESult:FREQuency:EVTCount? 2550
POWer:RIPPle:RESult:FREQuency:NPEak? 2550
POWer:RIPPle:RESult:FREQuency:PPEak? 2550
POWer:RIPPle:RESult:FREQuency:RMS? 2550
POWer:RIPPle:RESult:FREQuency:STDDev? 2550
POWer:RIPPle:RESult:FREQuency:WFMCount? 2550
POWer:RIPPle:RESult:MAXimum[:ACTual]? 2550
POWer:RIPPle:RESult:MAXimum:AVG? 2550
POWer:RIPPle:RESult:MAXimum:EVTCount? 2550
POWer:RIPPle:RESult:MAXimum:NPEak? 2550
POWer:RIPPle:RESult:MAXimum:PPEak? 2550
POWer:RIPPle:RESult:MAXimum:RMS? 2550
POWer:RIPPle:RESult:MAXimum:STDDev? 2550
POWer:RIPPle:RESult:MAXimum:WFMCount? 2550
POWer:RIPPle:RESult:MINimum[:ACTual]? 2551
POWer:RIPPle:RESult:MINimum:AVG? 2551
POWer:RIPPle:RESult:MINimum:EVTCount? 2551
POWer:RIPPle:RESult:MINimum:NPEak? 2551
POWer:RIPPle:RESult:MINimum:PPEak? 2551
POWer:RIPPle:RESult:MINimum:RMS? 2551
POWer:RIPPle:RESult:MINimum:STDDev? 2551
POWer:RIPPle:RESult:MINimum:WFMCount? 2551
POWer:RIPPle:RESult:NDCYcle[:ACTual]? 2551
POWer:RIPPle:RESult:NDCYcle:AVG? 2551
POWer:RIPPle:RESult:NDCYcle:EVTCount? 2551
POWer:RIPPle:RESult:NDCYcle:NPEak? 2551
POWer:RIPPle:RESult:NDCYcle:PPEak? 2551
POWer:RIPPle:RESult:NDCYcle:RMS? 2551
POWer:RIPPle:RESult:NDCYcle:STDDev? 2552
POWer:RIPPle:RESult:NDCYcle:WFMCount? 2552
POWer:RIPPle:RESult:PDCYcle[:ACTual]? 2552
POWer:RIPPle:RESult:PDCYcle:AVG? 2552
POWer:RIPPle:RESult:PDCYcle:EVTCount? 2552
POWer:RIPPle:RESult:PDCYcle:NPEak? 2552
POWer:RIPPle:RESult:PDCYcle:PPEak? 2552
POWer:RIPPle:RESult:PDCYcle:RMS? 2552
POWer:RIPPle:RESult:PDCYcle:STDDev? 2552
POWer:RIPPle:RESult:PDCYcle:WFMCount? 2552
POWer:RIPPle:RESult:PDEL[:ACTual]? 2553
POWer:RIPPle:RESult:PDEL:AVG? 2553
POWer:RIPPle:RESult:PDEL:EVTCount? 2553
POWer:RIPPle:RESult:PDEL:NPEak? 2553
POWer:RIPPle:RESult:PDEL:PPEak? 2553
POWer:RIPPle:RESult:PDEL:RMS? 2553
POWer:RIPPle:RESult:PDEL:STDDev? 2553
POWer:RIPPle:RESult:PDEL:WFMCount? 2553

POWer:RIPPle:RESult:PERiod[:ACTual]?....2553
POWer:RIPPle:RESult:PERiod:AVG?....2553
POWer:RIPPle:RESult:PERiod:EVTCount?....2553
POWer:RIPPle:RESult:PERiod:NPEak?....2553
POWer:RIPPle:RESult:PERiod:PPEak?....2553
POWer:RIPPle:RESult:PERiod:RMS?....2553
POWer:RIPPle:RESult:PERiod:STDDev?....2553
POWer:RIPPle:RESult:PERiod:WFMCount?....2553
POWer:RIPPle:RESult:STDDev[:ACTual]?....2554
POWer:RIPPle:RESult:STDDev:AVG?....2554
POWer:RIPPle:RESult:STDDev:EVTCount?....2554
POWer:RIPPle:RESult:STDDev:NPEak?....2554
POWer:RIPPle:RESult:STDDev:PPEak?....2554
POWer:RIPPle:RESult:STDDev:RMS?....2554
POWer:RIPPle:RESult:STDDev:STDDev?....2554
POWer:RIPPle:RESult:STDDev:WFMCount?....2554

POWer:RIPPle:AUToscale <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale> AUTO | MANUAL

*RST: MANUAL

POWer:RIPPle:CURRent <TwoChMeas>

Activates or deactivates the input current.

Parameters:

<TwoChMeas> ON | OFF

*RST: ON

POWer:RIPPle:EXECute

Starts the ripple measurement.

Usage: Event

POWer:RIPPle:FREQuency <SmpsFrequency>

Sets the SMPS switching frequency.

Parameters:

<SmpsFrequency> Range: 1 to 100E+9
Increment: 1
*RST: 1E+6
Default unit: Hz

POWer:RIPPle:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:RIPPle:RESult:FREQuency[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:FREQuency:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:FREQuency:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:FREQuency:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:FREQuency:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:FREQuency:RMS? [<VoltageMeasType>]
POWer:RIPPle:RESult:FREQuency:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:FREQuency:WFMCount? [<VoltageMeasType>]

Return the specified statistic result of the frequency of the signal. The result is based on the period measurement.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:

<VoltageMeasType> VOLTage | CURRent
*RST: VOLTage

Return values:

<Result> Statistic result of the frequency

Usage: Query only

POWer:RIPPle:RESult:MAXimum[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:MAXimum:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:MAXimum:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:MAXimum:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:MAXimum:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:MAXimum:RMS? [<VoltageMeasType>]
POWer:RIPPle:RESult:MAXimum:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:MAXimum:WFMCount? [<VoltageMeasType>]

Return the specified statistic result for the maximum value of the waveform.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results

- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<VoltageMeasType> VOLTage | CURRent
*RST: VOLTage

Return values:
<Result> Statistic result

Usage: Query only

POWer:RIPPle:RESult:MINimum[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:MINimum:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:MINimum:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:MINimum:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:MINimum:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:MINimum:RMS? [<VoltageMeasType>]
POWer:RIPPle:RESult:MINimum:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:MINimum:WFMCount? [<VoltageMeasType>]

Return the specified statistic result for the minimum value of the waveform.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<VoltageMeasType> VOLTage | CURRent
*RST: VOLTage

Return values:
<Result> Statistic result

Usage: Query only

POWer:RIPPle:RESult:NDCYcle[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:NDCYcle:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:NDCYcle:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:NDCYcle:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:NDCYcle:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:NDCYcle:RMS? [<VoltageMeasType>]

POWer:RIPPle:RESult:NDCYcle:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:NDCYcle:WFMCount? [<VoltageMeasType>]

Return the specified statistic result for the negative duty cycle. The measurement requires at least one complete period of a triggered signal.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:

<VoltageMeasType> VOLTage | CURRent

*RST: VOLTage

Return values:

<Result> Statistic result

Usage: Query only

POWer:RIPPle:RESult:PDCYcle[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDCYcle:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDCYcle:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDCYcle:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDCYcle:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDCYcle:RMS? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDCYcle:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDCYcle:WFMCount? [<VoltageMeasType>]

Return the specified statistic result for the positive duty cycle. The measurement requires at least one complete period of a triggered signal.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:

<VoltageMeasType> VOLTage | CURRent

*RST: VOLTage

Return values:

<Result> Statistic result

Usage: Query only

POWer:RIPPle:RESult:PDEL[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDEL:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDEL:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDEL:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDEL:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDEL:RMS? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDEL:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:PDEL:WFMCount? [<VoltageMeasType>]

Return the specified statistic result for the peak to peak measurement.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:

<VoltageMeasType> VOLTage | CURRent
*RST: VOLTage

Return values:

<Result> Statistic result

Usage: Query only

POWer:RIPPle:RESult:PERiod[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:PERiod:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:PERiod:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:PERiod:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:PERiod:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:PERiod:RMS? [<VoltageMeasType>]
POWer:RIPPle:RESult:PERiod:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:PERiod:WFMCount? [<VoltageMeasType>]

Return the specified statistic result for the period, the length of the left-most signal period of the waveform.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<VoltageMeasType> VOLTage | CURRent
*RST: VOLTage

Return values:
<Result> Statistic result

Usage: Query only

POWer:RIPPle:RESult:STDDev[:ACTual]? [<VoltageMeasType>]
POWer:RIPPle:RESult:STDDev:AVG? [<VoltageMeasType>]
POWer:RIPPle:RESult:STDDev:EVTCount? [<VoltageMeasType>]
POWer:RIPPle:RESult:STDDev:NPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:STDDev:PPEak? [<VoltageMeasType>]
POWer:RIPPle:RESult:STDDev:RMS? [<VoltageMeasType>]
POWer:RIPPle:RESult:STDDev:STDDev? [<VoltageMeasType>]
POWer:RIPPle:RESult:STDDev:WFMCount? [<VoltageMeasType>]

Return the specified statistic result for the standard deviation of the long-term measurement results.

- [:ACTual]: current measurement result
- AVG: average of the long-term measurement results
- EVTCount: number of measurement results in the long-term measurement
- NPEak: negative peak value of the long-term measurement results
- PPEak: positive peak value of the long-term measurement results
- RMS: RMS value of the long-term measurement results
- STDDev: standard deviation of the long-term measurement results

Query parameters:
<VoltageMeasType> VOLTage | CURRent
*RST: VOLTage

Return values:
<Result> Statistic result

Usage: Query only

24.23.15 Transient response

This measurement is a single shot measurement. To start the measurement, use the `RUNS` command.

Programming example: Chapter 24.3.8.2, "Transient response measurement", on page 1278

POWer:TRANsient:AUToscale....2555
POWer:TRANsient:EXECute....2555
POWer:TRANsient:FREQuency....2555
POWer:TRANsient:HYSTeresis....2555
POWer:TRANsient:INPut....2555

POWer:TRANsient:REPort:ADD....2556
POWer:TRANsient:RESult[:ACTual]?....2556
POWer:TRANsient:SIGHigh....2556
POWer:TRANsient:SIGLow....2556
POWer:TRANsient:TRGChannel....2556
POWer:TRANsient:TRGLevel....2557
POWer:TRANsient:TRGSlope....2557

POWer:TRANsient:AUToscale <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale> AUTO | MANUAL

*RST: MANUAL

POWer:TRANsient:EXECute

Starts the transient response measurement.

Usage: Event

POWer:TRANsient:FREQuency <SmpsFrequency>

Sets the SMPS switching frequency.

Parameters:

<SmpsFrequency>
Range: 1 to 100E+6
Increment: 1
*RST: 1E+6
Default unit: Hz

POWer:TRANsient:HYSTeresis <ToleranceTube>

Specifies a tolerated error band for the signal level.

Parameters:

<ToleranceTube>
Range: 0 to 50
Increment: 1
*RST: 10
Default unit: %

POWer:TRANsient:INPut <ThreeChMeas>

Activates or deactivates the input voltage.

Parameters:

<ThreeChMeas> ON | OFF

*RST: OFF

POWer:TRANsient:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:TRANsient:RESult[:ACTual]? <MeasType>

Returns the results of the transient response measurement.

Query parameters:

<MeasType> OVERshoot | RTIMe | DELay | PEAKtime | SETTling

Usage: Query only

POWer:TRANsient:SIGHigh <ExpHighOutputSignLev>

Sets the expected signal high voltage value.

Parameters:

<ExpHighOutputSignLev>
Range: -1000 to 1000
Increment: 1E-3
*RST: 1
Default unit: V

POWer:TRANsient:SIGLow <ExpLowOutputSignLev>

Sets the expected signal low voltage value.

Parameters:

<ExpLowOutputSignLev>
Range: -1000 to 1000
Increment: 1E-3
*RST: 0
Default unit: V

POWer:TRANsient:TRGChannel <TriggerSource>

Sets the source channel of the trigger.

Parameters:

<TriggerSource> CHAN1 | CHANnel1 | CHAN2 | CHANnel2 | CHAN3 | CHANnel3 | CHAN4 | CHANnel4

CHAN1 = CHANnel1, CHAN2 = CHANnel2, CHAN3 = CHANnel3, CHAN4 = CHANnel4
Only the measurement source channels can be used as trigger source.

*RST: CHAN1

POWer:TRANsient:TRGLevel <TriggerLevel>

Parameters:

<TriggerLevel>	Range:	-1000 to 1000
	Increment:	1E-3
	*RST:	0
	Default unit:	V

POWer:TRANsient:TRGSlope <TriggerSlope>

Sets the edge type for the trigger event.

Parameters:

<TriggerSlope>	POSitive | NEGative | EITHer	
	*RST:	POSitive

24.23.16 Spectrum

POWer:SPECtrum:AUToscale....2557
POWer:SPECtrum:EXECute....2557
POWer:SPECtrum:FREQuency....2557
POWer:SPECtrum:REPort:ADD....2558
POWer:SPECtrum:RCOunt?....2558
POWer:SPECtrum:RESult<m>:FREQuency?....2558
POWer:SPECtrum:RESult<m>:LEVel?....2558

POWer:SPECtrum:AUToscale <Autoscale>

Selects the scaling for the display of the results.

Parameters:

<Autoscale>	AUTO | MANUAL	
	*RST:	MANUAL

POWer:SPECtrum:EXECute

Starts the spectrum measurement.

Usage: Event

POWer:SPECtrum:FREQuency <SmpsFrequency>

Sets the SMPS switching frequency.

Parameters:

<SmpsFrequency>	Range:	1 to 100E+6
	Increment:	1
	*RST:	1E+6
	Default unit:	Hz

POWer:SPECtrum:REPort:ADD

Adds the result to the report list.

Usage: Event

POWer:SPECtrum:RCOunt?

Returns the total number of harmonics.

Return values:
<ResultCount>

Usage: Query only

POWer:SPECtrum:RESult<m>:FREQuency?

Returns the result frequency of the m-th result value.

Suffix:
<m> 1..*

Return values:
<Frequency>

Usage: Query only

POWer:SPECtrum:RESult<m>:LEVel?

Returns the result level of the m-th result value.

Suffix:
<m> 1..*

Return values:
<Level>

Usage: Query only

24.24 Deembedding (option R&S RTO6-K121)

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"

- Chapter 24.3, "Programming examples", on page 1259

- Measurement setup 2559
- Save and load the deembedding setup 2561
- Components 2562
- Responses 2568

24.24.1 Measurement setup

DEEMbedding<m>[:STATe] 2559
DEEMbedding<m>:ADD 2559
DEEMbedding<m>:REMove 2560
DEEMbedding<m>:CCOunt? 2560
DEEMbedding<m>:BANDwidth 2560
DEEMbedding<m>:CMPDelay 2561
DEEMbedding<m>:PRESet 2561

DEEMbedding<m>[:STATe] <State>

Activates the deembedding - the correction of parasitic effects of the measurement setup on the measured signal.

Suffix:

<m> 1..4
Selects the channel that is configured for deembedding.

Parameters:

<State> ON | OFF
*RST: OFF

Usage: Asynchronous command

DEEMbedding<m>:ADD <Index>, [<Type>]

Adds a component to the measurement setup.

Suffix:

<m> 1..4
Selects the channel that is configured for deembedding.

Setting parameters:

<Index> Defines the place of the component in the measurement path. The DUT is always the first component (index = 1), the probe or input is the last component.

<Type> PROBe | FIXTure | CABLe | DUT | ADAPter | INPut | OTHer | RTZA10 | RTZA16 | RTZA17 | PP | PC | COPRobes

RTZA10 | RTZA16 | RTZA17
SMA adapters and cable offered by Rohde & Schwarz

PP | PC
Proven probe, proven cable

Example: See Chapter 24.3.6.1, "Adding cables to the measurement setup", on page 1273.

Usage: Setting only
Asynchronous command

DEEMbedding<m>:REMove <Index>

Deletes the component with the indicated index from the measurement setup.

Suffix:
<m> 1..4
Selects the channel that is configured for deembedding.

Setting parameters:
<Index> Defines the place of the component in the measurement path. The DUT is always the first component (index = 1), the probe or input is the last component.

Usage: Setting only
Asynchronous command

DEEMbedding<m>:CCOunt?

Returns the total number of components in the measurement setup, including disabled components.

Suffix:
<m> 1..4
Selects the channel that is configured for deembedding.

Return values:
<Value> Integer value, number of components

Usage: Query only

DEEMbedding<m>:BANDwidth <Bandwidth>

Sets the maximum bandwidth until which the signal is corrected. This maximum value is the minimum bandwidth value of probe, probe tip and oscilloscope bandwidth, and can not be higher than the highest frequency in a used S-parameter file.

Suffix:
<m> 1..4
Selects the channel that is configured for deembedding.

Parameters:
<Bandwidth> Bandwidth limit for correction.

Range: 1E+9 to 16E+9
Increment: 10
*RST: 4E+9
Default unit: Hz

Usage: Asynchronous command

DEEMbedding<m>:CMPDelay <DelayCompens>

Enables the compensation for the group delay of the complete measurement setup. If enabled, the calibration time reference point is the start point of the measurement setup. If disabled, the calibration time reference point is the oscilloscope's channel input.

Suffix:
<m> 1..4
Selects the channel that is configured for deembedding.

Parameters:
<DelayCompens> ON | OFF
*RST: ON

Usage: Asynchronous command

DEEMbedding<m>:PRESet

Sets the complete deembedding configuration to the default values. Use this command always before setting up the deembedding, because *RST does not affect the deembedding settings.

Suffix:
<m> 1..4
Selects the channel that is configured for deembedding.

Example: See Chapter 24.3.6.1, "Adding cables to the measurement setup", on page 1273.

Usage: Event
Asynchronous command

24.24.2 Save and load the deembedding setup

DEEMbedding<m>:NAME....2561
DEEMbedding<m>:SAVE....2562
DEEMbedding<m>:OPEN....2562
DEEMbedding<m>:DELete....2562

DEEMbedding<m>:NAME <Path>

Sets the file name, file format and path of the deembedding setup file. The file format is always `.xml`.

Suffix:
<m> 1..4
Selects the channel that is configured for deembedding.

Parameters:
<Path> String with path and file name with extension `.xml`

Usage: Asynchronous command

DEEMbedding<m>:SAVE

Saves the deembedding configuration to the file specified with `DEEMbedding<m>:NAME`.

Suffix:
<m> 1..4
Selects the channel that is configured for deembedding.

Usage: Event

DEEMbedding<m>:OPEN

Opens and loads the deembedding configuration from the file specified with `DEEMbedding<m>:NAME`.

Suffix:
<m> 1..4
Selects the channel that is configured for deembedding.

Usage: Event

DEEMbedding<m>:DELete

Deletes the deembedding configuration file specified with `DEEMbedding<m>:NAME`.

Suffix:
<m> 1..4
Selects the channel that is configured for deembedding.

Usage: Event

24.24.3 Components

- General settings....2562
- DUT settings....2564
- R&S RT-ZM probe settings....2565
- Cable, adapter, fixture and custom settings....2567

24.24.3.1 General settings

DEEMbedding<m>:COMPonent<n>[:STATe] <IsEnabled>

Enables or disables the indicated component for deembedding.

The first and the last component are always enabled and cannot be disabled.

Suffix:

<m> 1..4
Selects the channel that is configured for deembedding.

<n> 2..n-1
Index of the component in the measurement setup. The DUT has index 1, the probe has index n.

Parameters:

<IsEnabled> ON | OFF

*RST: OFF

Usage: Asynchronous command

DEEMbedding<m>:COMPonent<n>:TYPE <Type>

Sets the type of the indicated component. The first component is always DUT, the last component is PROBe or INPut.

Suffix:

<m> 1..4
Selects the channel that is configured for deembedding.

<n> 1..n
Index of the component in the measurement setup. The DUT has index 1, the probe has index n.

Parameters:

<Type> PROBe | FIXTure | CABLe | DUT | ADAPter | INPut | OTHer | RTZA10 | RTZA16 | RTZA17 | PP | PC

RTZA10 | RTZA16 | RTZA17
SMA adapters and cable offered by Rohde & Schwarz

PP | PC
Proven probe, proven cable

*RST: CABLe

Usage: Asynchronous command

DEEMbedding<m>:COMPonent<n>:NAME <Name>

Sets a name for the indicated component that helps you identifying it.

Suffix:

<m> 1..4
Selects the channel that is configured for deembedding.

<n> 1..n
Index of the component in the measurement setup. The DUT has index 1, the probe has index n.

Parameters:

<Name> String parameter, name of the component

Usage: Asynchronous command

DEEMbedding<m>:COMPonent<n>:PCOunt?

Returns the number of all ports of the selected component (input ports and output ports).

The number of ports depends on the attached probe. If a modular probe R&S RT-ZM or differential probe R&S RT-ZD is used, the number of ports is 4. Otherwise, 2 ports are available.

Suffix:

<m> 1..4
Selects the channel that is configured for deembedding.

<n> 1..n
Index of the component in the measurement setup. The DUT has index 1, the probe has index n.

Return values:

<TotalPorts> ONE | TWO | THRee | FOUR

*RST: TWO

Usage: Query only
Asynchronous command

24.24.3.2 DUT settings

DEEMbedding<m>:COMPonent<n>:MODE....2564
DEEMbedding<m>:COMPonent<n>:LOAD:REMProbeload....2565
DEEMbedding<m>:COMPonent<n>:LOAD:IMPedance?....2565

DEEMbedding<m>:COMPonent<n>:MODE <Mode>

Sets the probing mode.

Suffix:

<m> 1..4
Selects the channel that is configured for deembedding.

<n> 1, only for available DUT

Parameters:

<Mode> PROBing | TERMinating

PROBing
For measurements with a high-ohmic probe and measure on an existing line in parallel to the load.

TERMinating
For measurements if the measuring equipment is the load of the line.

*RST: PROBing

Usage: Asynchronous command

DEEMbedding<m>:COMPonent<n>:LOAD:REMProbeload <Value>

Removes the loading of the probe if `DEEMbedding<m>:COMPonent<n>:MODE` is set to `PROBing`.

If the probe loading is removed, you see the signal that would be at the measurement point if the probe's input impedance is ideal (infinite impedance).

Without removing the loading, you see the real signal at the measurement point, including the probe loading.

Suffix:
<m> 1..4
Selects the channel that is configured for deembedding.

<n> 1, only for available DUT

Parameters:
<Value> OFF | ON

Usage: Asynchronous command

DEEMbedding<m>:COMPonent<n>:LOAD:IMPedance? <Value>

Shows the source impedance of the DUT. If an R&S RT-ZM probe is connected, the value depends on the selected probe mode: common mode, differential, or single-ended measurement.

Suffix:
<m> 1..4
Selects the channel that is configured for deembedding.

<n> 1, only for available DUT

Return values:
<Value> Numeric value

Usage: Query only
Asynchronous command

24.24.3.3 R&S RT-ZM probe settings

This chapter lists the commands that correspond to the "Deembedding" > "RT-ZMxx" dialog box. More commands are provided in Chapter 24.8.7.6, "Modular probes, deembedding", on page 1358.

PROBe<m>:DEEMbedding:TIPModule[:SELect] 2566
PROBe<m>:DEEMbedding:TIPModule:ZMA<n>:SUBModule 2566
PROBe<m>:DEEMbedding:BANDwidth 2566

PROBe<m>:DEEMbedding:TIPModule[:SELect] <ProbeTip>

Selects the tip module that is used for measurement.

Suffix:
<m> 1..4
Selects the channel to which the probe is connected.

Parameters:
<ProbeTip> NONE | ZMA10 | ZMA12 | ZMA14 | ZMA15 | ZMA30 | ZMA40 | ZMA50 | CUST

*RST: ZMA10

Usage: Asynchronous command

PROBe<m>:DEEMbedding:TIPModule:ZMA<n>:SUBModule <Value>

Defines additional settings for some tip modules:

For the browser module R&S RT-ZMA30, measure the space between the pins, and select the appropriate value.

If R&S RT-ZMA40 is selected, choose the used submodule: semi-rigid cables or none.

Suffix:
<m> 1..4
Selects the channel to which the probe is connected.

<n> 30 (R&S RT-ZMA30) or 40 (R&S RT-ZMA40)

Parameters:
<Value> D16 | D25 | D45 | D75 | NONE | SRCS

For R&S RT-ZMA30:
D16 = 1.6 mm
D25 = 2.5 mm
D45 = 4.5 mm
D75 = 7.5 mm
For R&S RT-ZMA40:
NONE = no submodule
SRCS = semi-rigid cables

Usage: Asynchronous command

PROBe<m>:DEEMbedding:BANDwidth <ProbeBandwidth>

Sets the maximum bandwidth until which the signal is corrected. This maximum value is the minimum bandwidth value of probe, probe tip and oscilloscope bandwidth, and can not be higher than the highest frequency in a used S-parameter file. Consider that most tip modules support the full bandwidth of the probe amplifier, but some tip modules have limited bandwidth.

Suffix:
<m> 1..4
Selects the channel to which the probe is connected.

Parameters:
<ProbeBandwidth> Range: 10000 to 16E+9
Increment: 10
*RST: 4E+9
Default unit: Hz

Usage: Asynchronous command

24.24.3.4 Cable, adapter, fixture and custom settings

DEEMbedding<m>:COMPonent<n>:IDENtical 2567
DEEMbedding<m>:COMPonent<n>:IPONe 2567
DEEMbedding<m>:COMPonent<n>:IPTWo 2567
DEEMbedding<m>:COMPonent<n>:OPONe 2567
DEEMbedding<m>:COMPonent<n>:OPTWo 2567
DEEMbedding<m>:COMPonent<n>:SPONe 2568
DEEMbedding<m>:COMPonent<n>:SPTWo 2568

DEEMbedding<m>:COMPonent<n>:IDENtical <MatchedPaths>

The setting is relevant if the measurement setup is a 4-port, and the component is a 2-port. Usually, these components are cables and adapters, which are described by *.s2p files. In case of a 4-port setup, 2 cables or 2 adapters are used. If they are identical, for example, when a matched-pair cable is used, they can be described by the same file. In this case, set the command to ON.

Suffix:
<m> 1..4
Selects the channel that is configured for deembedding.

<n> 1..*
Selects the component. The DUT has suffix 1.

Parameters:
<MatchedPaths> ON | OFF
*RST: OFF

Usage: Asynchronous command

DEEMbedding<m>:COMPonent<n>:IPONe <InputPort1>
DEEMbedding<m>:COMPonent<n>:IPTWo <InputPort2>
DEEMbedding<m>:COMPonent<n>:OPONe <OutputPort1>
DEEMbedding<m>:COMPonent<n>:OPTWo <OutputPort2>

Assign the ports to the input and output of the component according to the data in the Touchstone file. This information is not directly written in the file. You need background information how the S-parameters were determined, i.e. which port was input and which port was output.

Suffix:

<m> 1..4
Selects the channel that is configured for deembedding.

<n> 1..*
Selects the component. The DUT has suffix 1.

Parameters:

<InputPort1>
<InputPort2>
<OutputPort1>
<OutputPort2>

ONE | TWO | THRee | FOUR

Port number that is assigned to the input or output.

*RST: ONE for input 1, TWO for output 1, THRee for input 2, FOUR for output 4

Usage: Asynchronous command

DEEMbedding<m>:COMPonent<n>:SPONe <SParamFilePath>
DEEMbedding<m>:COMPonent<n>:SPTWo <SParamFilePath>

Set the file name, file format and path of the S-parameter file, and load the S-parameters.

See also: "S-Parameters, Filetype" on page 1208.

Suffix:

<m> 1..4
Selects the channel that is configured for deembedding.

<n> 1..*
Selects the component. The DUT has suffix 1.

Parameters:

<SParamFilePath> String with path and file name with extension `s2p` (2-port component) or `s4p` (4-port component).

Example: See Chapter 24.3.6.1, "Adding cables to the measurement setup", on page 1273.

Usage: Asynchronous command

24.24.4 Responses

DEEMbedding<m>:CADC?

Maximum attenuation or gain of the measurement setup before deembedding at DC. This value is the expected basic attenuation or gain, which is corrected by deembedding.

Suffix:

<m> 1..4
Selects the channel that is configured for deembedding.

Return values:

ValueAtDC
Range: -1E+9 to 1E+9
Increment: 0.01
*RST: 0
Default unit: dB

Usage: Query only

DEEMbedding<m>:MATTenuation?

Maximum attenuation of the measurement setup before deembedding, with reference to a frequency response normalized at DC to 0 dB. The deembedding filter must amplify the amplitude response by the "Att_{max}" value, and thus also increases the noise by this value. The value also helps to decide whether deembedding is useful for the defined effective bandwidth.

Suffix:

<m>
1..4
Selects the channel that is configured for deembedding.

Return values:

<MaxFilterAtt>
Range: -1E+9 to 1E+9
Increment: 0.01
*RST: 0
Default unit: dB

Usage: Query only

DEEMbedding<m>:MGAin?

Maximum gain of the measurement setup before deembedding, with reference to a frequency response normalized at DC to 0 dB. The deembedding filter must attenuate the amplitude response by this value. The value helps also to avoid an overload of the ADC.

Suffix:

<m>
1..4
Selects the channel that is configured for deembedding.

Return values:

<MaxFilterGn>
Range: -1E+9 to 1E+9
Increment: 0.01
*RST: 0
Default unit: dB

Usage: Query only

24.25 Maintenance

Some of the commands in the following chapter are asynchronous. An overlapping or asynchronous command does not automatically finish executing before the next command starts executing. If overlapping commands must be executed in a defined order, e.g. to avoid wrong measurement results, they must be serviced sequentially.

To prevent an overlapping execution of commands, one of the commands `*OPC`, `*OPC?` or `*WAI` can be used after the command or a command set.

For more information, see:

- www.rohde-schwarz.com/rc-via-scpi, chapter "Command Sequence and Synchronization"
- Chapter 24.3, "Programming examples", on page 1259

DIAGnostic:SERVice:WFAModel?....2570
DIAGnostic:SERVice:WFASeries?....2570
DIAGnostic:SERVice:WFAType?....2570
CALibration:DATE?....2571
CALibration:TIME?....2571
CALibration:RESult?....2571
DIAGnostic:SERVice:STST:EXECute....2571
DIAGnostic:SERVice:STST:STATe?....2571
DIAGnostic:SERVice:PWD....2572

DIAGnostic:SERVice:WFAModel?

Returns the model name of the oscilloscope.

Return values:

<WFAModel> Product type

Usage: Query only
Asynchronous command

DIAGnostic:SERVice:WFASeries?

Returns the model series of the oscilloscope.

Return values:

<WFASeries> RTO | RTE | RTP

*RST: RTO

Usage: Query only
Asynchronous command

DIAGnostic:SERVice:WFAType?

Returns the instrument family of the oscilloscope.

Return values:
<WFAType> RTO2000 | RTO6 | RTP | RTPFL

RTP: part numbers 1320.5007
RTPFI: part numbers 1803.7000

*RST: RTO2000

Usage: Query only
Asynchronous command

CALibration:DATE?

Returns the date of the last selfalignment.

Return values:
<Date>

Usage: Query only
Asynchronous command

CALibration:TIME?

Returns the time of the last selfalignment.

Return values:
<Time>

Usage: Query only
Asynchronous command

CALibration:RESult?

Returns the result of the last selfalignment and the current alignment status. In remote mode, `*CAL?` provides more detailed information.

Return values:
<ResultState> PASSed | FAILed | NOALigndata

*RST: FAILed

Usage: Query only
Asynchronous command

DIAGnostic:SERVice:STST:EXECute

Starts the selftest.

Usage: Event
Asynchronous command

DIAGnostic:SERVice:STST:STATe?

Returns the summary result of the selftest.

Return values:
<State> PSSD | FAILed | UNDefined
*RST: UNDefined

Usage: Query only
Asynchronous command

DIAGnostic:SERVice:PWD <Password>

Sets the password to enter the service mode.

Setting parameters:
<Password> Password string

Usage: Setting only
Asynchronous command

24.26 Status reporting

This chapter describes the remote commands that are used to read the status registers.

For information on structure, hierarchy, and contents of the status registers, see Chapter 22.5.4, "Remote control - status reporting system", on page 1245.

- STATus:OPERation register....2572
- STATus:QUEStionable registers....2573

24.26.1 STATus:OPERation register

`STATus:OPERation` commands provide information on the activity of the instrument.

See also: "STATus:OPERation register" on page 1246

STATus:OPERation:CONDition?....2572
STATus:OPERation[:EVENt]?....2572
STATus:OPERation:ENABle....2573

STATus:OPERation:CONDition?
STATus:OPERation[:EVENt]?

The CONDition command returns information on actions the instrument is currently executing. The contents of the register is retained.

The EVENt command returns information on actions the instrument has executed since the last reading. Reading the EVENt register deletes its contents.

Bits:

- 0 = ALIGnment
- 2 = AUToset

- 3= WTRIgger (wait for trigger)
- 4= MEASuring

Usage: Query only

STATus:OPERation:ENABle <Enable>

Controls the ENABle part of the STATus:OPERation register. The ENABle defines which events in the EVENt part of the status register are forwarded to the OPERation summary bit (bit 7) of the status byte. The status byte can be used to create a service request.

Parameters:

<Enable> Range: 1 to 65535
Increment: 1

Example: `STATus:OPERation:ENABle 5`
The ALIGnment event (bit 0) and AUToset event (bit 2) are forwarded to the OPERation summary bit of the status byte.

24.26.2 STATus:QUEStionable registers

The commands of the `STATus:QUEStionable` subsystem control the status reporting structures of the STATus:QUEStionable registers.

See also: "STATus:QUEStionable register" on page 1247

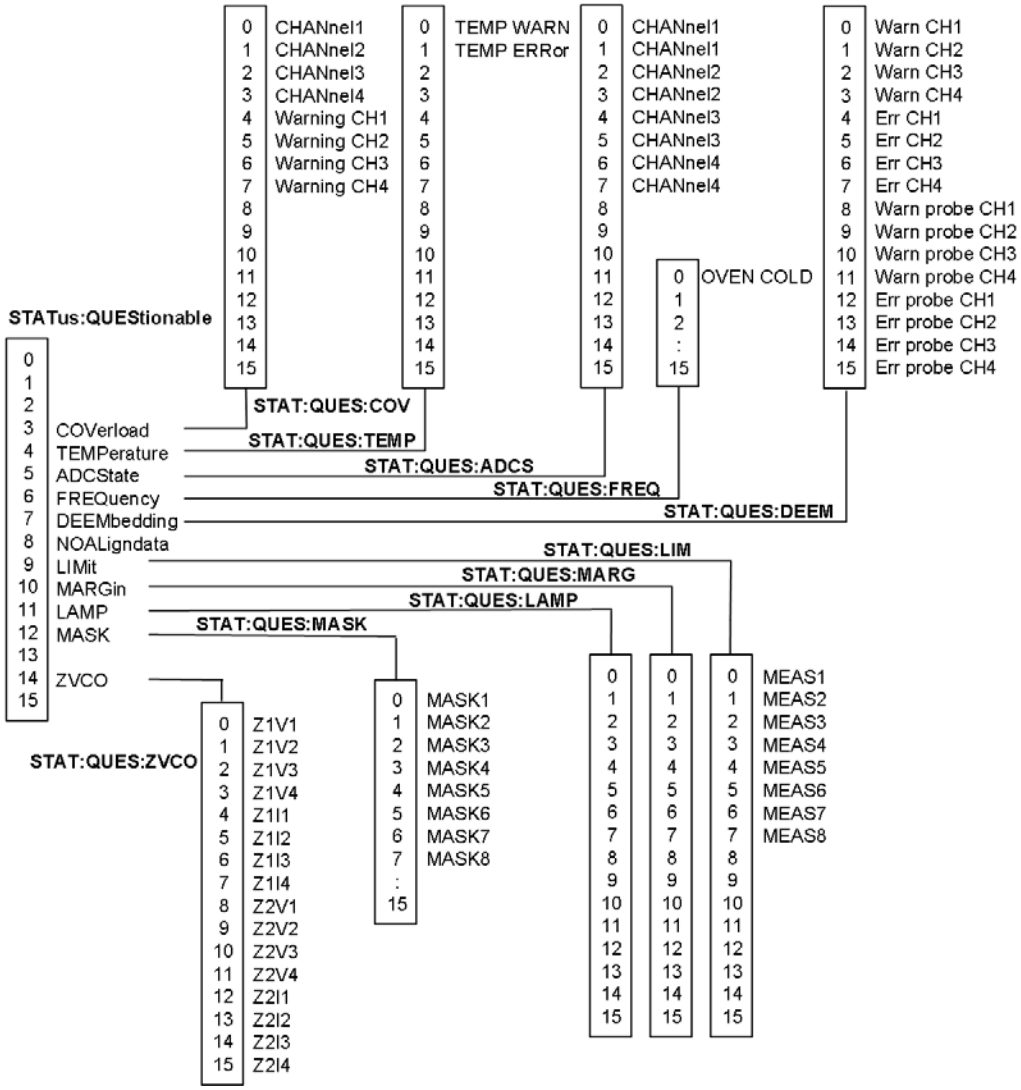


Figure 24-1: Overview of the STATus:QUEStionable register

The following commands are available:

STATus:QUEStionable:COVerload:CONDition?....2575
STATus:QUEStionable:TEMPerature:CONDition?.... 2575
STATus:QUEStionable:ADCState:CONDition?.... 2575
STATus:QUEStionable:LIMit:CONDition?....2575
STATus:QUEStionable:MARGin:CONDition?.... 2575
STATus:QUEStionable:FREQuency:CONDition?....2575
STATus:QUEStionable:DEEMbedding:CONDition?....2575
STATus:QUEStionable:LAMPlitude:CONDition?.... 2575
STATus:QUEStionable:ZVCoverload:CONDition?....2576
STATus:QUEStionable:MASK:CONDition?.... 2576
STATus:QUEStionable:COVerload:ENABle....2576
STATus:QUEStionable:TEMPerature:ENABle.... 2576
STATus:QUEStionable:ADCState:ENABle.... 2576
STATus:QUEStionable:LIMit:ENABle....2576

STATus:QUEStionable:MARGin:ENABle ... 2576
STATus:QUEStionable:FREQuency:ENABle ... 2576
STATus:QUEStionable:LAMPlitude:ENABle ... 2576
STATus:QUEStionable:ZVCoverload:ENABle ... 2576
STATus:QUEStionable:MASK:ENABle ... 2576
STATus:QUEStionable:COVerload[:EVENt]? ... 2576
STATus:QUEStionable:TEMPerature[:EVENt]? ... 2576
STATus:QUEStionable:ADCState[:EVENt]? ... 2576
STATus:QUEStionable:LIMit[:EVENt]? ... 2576
STATus:QUEStionable:MARGin[:EVENt]? ... 2576
STATus:QUEStionable:FREQuency:EVENt? ... 2576
STATus:QUEStionable:DEEMbedding[:EVENt]? ... 2576
STATus:QUEStionable:LAMPlitude[:EVENt]? ... 2576
STATus:QUEStionable:ZVCoverload[:EVENt]? ... 2576
STATus:QUEStionable:MASK[:EVENt]? ... 2576
STATus:QUEStionable:COVerload:NTRansition ... 2576
STATus:QUEStionable:TEMPerature:NTRansition ... 2576
STATus:QUEStionable:ADCState:NTRansition ... 2576
STATus:QUEStionable:LIMit:NTRansition ... 2577
STATus:QUEStionable:MARGin:NTRansition ... 2577
STATus:QUEStionable:FREQuency:NTRansition ... 2577
STATus:QUEStionable:DEEMbedding:NTRansition ... 2577
STATus:QUEStionable:LAMPlitude:NTRansition ... 2577
STATus:QUEStionable:ZVCoverload:NTRansition ... 2577
STATus:QUEStionable:MASK:NTRansition ... 2577
STATus:QUEStionable:COVerload:PTRansition ... 2577
STATus:QUEStionable:TEMPerature:PTRansition ... 2577
STATus:QUEStionable:ADCState:PTRansition ... 2577
STATus:QUEStionable:LIMit:PTRansition ... 2577
STATus:QUEStionable:MARGin:PTRansition ... 2577
STATus:QUEStionable:FREQuency:PTRansition ... 2577
STATus:QUEStionable:DEEMbedding:PTRansition ... 2577
STATus:QUEStionable:LAMPlitude:PTRansition ... 2577
STATus:QUEStionable:ZVCoverload:PTRansition ... 2577
STATus:QUEStionable:MASK:PTRansition ... 2577
MTESt:SBITnumber? ... 2577

STATus:QUEStionable:COVerload:CONDition?
STATus:QUEStionable:TEMPerature:CONDition?
STATus:QUEStionable:ADCState:CONDition?
STATus:QUEStionable:LIMit:CONDition?
STATus:QUEStionable:MARGin:CONDition?
STATus:QUEStionable:FREQuency:CONDition?
STATus:QUEStionable:DEEMbedding:CONDition?
STATus:QUEStionable:LAMPlitude:CONDition?

STATus:QUEStionable:ZVCoverload:CONDition?
STATus:QUEStionable:MASK:CONDition?

Returns the contents of the CONDition part of the status register to check for questionable instrument or measurement states. Reading the CONDition registers does not delete the contents.

Usage: Query only

STATus:QUEStionable:COVerload:ENABle <Value>
STATus:QUEStionable:TEMPerature:ENABle <Value>
STATus:QUEStionable:ADCState:ENABle <Value>
STATus:QUEStionable:LIMit:ENABle <Value>
STATus:QUEStionable:MARGin:ENABle <Value>
STATus:QUEStionable:FREQuency:ENABle <Value>
STATus:QUEStionable:LAMPlitude:ENABle <Value>
STATus:QUEStionable:ZVCoverload:ENABle <Value>
STATus:QUEStionable:MASK:ENABle <Value>

Sets the ENABle part that allows true conditions in the EVENt part to be reported in the summary bit. If a bit is set to 1 in the enable part and its associated event bit transitions to true, a positive transition occurs in the summary bit and is reported to the next higher level.

Parameters:
<Value> Bit mask in decimal representation

Example: `STATus:QUEStionable:MASK:ENABle 24`
Set bits no. 3 and 4 of the STATus:QUEStionable:MASK:ENABle register part: $24 = 8 + 16 = 2^3 + 2^4$

STATus:QUEStionable:COVerload[:EVENt]?
STATus:QUEStionable:TEMPerature[:EVENt]?
STATus:QUEStionable:ADCState[:EVENt]?
STATus:QUEStionable:LIMit[:EVENt]?
STATus:QUEStionable:MARGin[:EVENt]?
STATus:QUEStionable:FREQuency:EVENt?
STATus:QUEStionable:DEEMbedding[:EVENt]?
STATus:QUEStionable:LAMPlitude[:EVENt]?
STATus:QUEStionable:ZVCoverload[:EVENt]?
STATus:QUEStionable:MASK[:EVENt]?

Returns the contents of the EVENt part of the status register to check whether an event has occurred since the last reading. Reading an EVENt register deletes its contents.

Usage: Query only

STATus:QUEStionable:COVerload:NTRansition <Value>
STATus:QUEStionable:TEMPerature:NTRansition <Value>
STATus:QUEStionable:ADCState:NTRansition <Value>

STATus:QUEStionable:LIMit:NTRansition <Value>
STATus:QUEStionable:MARGin:NTRansition <Value>
STATus:QUEStionable:FREQuency:NTRansition <Value>
STATus:QUEStionable:DEEMbedding:NTRansition <Value>
STATus:QUEStionable:LAMPlitude:NTRansition <Value>
STATus:QUEStionable:ZVCoverload:NTRansition <Value>
STATus:QUEStionable:MASK:NTRansition <Value>

Sets the negative transition filter. If a bit is set, a 1 to 0 transition in the corresponding bit of the condition register causes a 1 to be written in the corresponding bit of the event register.

Parameters:
<Value> Bit mask in decimal representation

Example: `STATus:QUEStionable:MASK:NTRansition 24`
Set bits no. 3 and 4 of the STATus:QUEStionable:MASK:NTRansition register part: $24 = 8 + 16 = 2^3 + 2^4$

STATus:QUEStionable:COVerload:PTRansition <Value>
STATus:QUEStionable:TEMPerature:PTRansition <Value>
STATus:QUEStionable:ADCState:PTRansition <Value>
STATus:QUEStionable:LIMit:PTRansition <Value>
STATus:QUEStionable:MARGin:PTRansition <Value>
STATus:QUEStionable:FREQuency:PTRansition <Value>
STATus:QUEStionable:DEEMbedding:PTRansition <Value>
STATus:QUEStionable:LAMPlitude:PTRansition <Value>
STATus:QUEStionable:ZVCoverload:PTRansition <Value>
STATus:QUEStionable:MASK:PTRansition <Value>

Sets the positive transition filter. If a bit is set, a 0 to 1 transition in the corresponding bit of the condition register causes a 1 to be written in the corresponding bit of the event register.

Parameters:
<Value> Bit mask in decimal representation

Example: `STATus:QUEStionable:MASK:PTRansition 24`
Set bits no. 3 and 4 of the STATus:QUEStionable:MASK:PTRansition register part: $24 = 8 + 16 = 2^3 + 2^4$

MTESt:SBITnumber? <MaskTestName>

Returns the number of the status bit that belongs to the indicated mask test in the `STATus:QUEStionable:MASK` register. The status bits are assigned in the order of mask creation.

Query parameters:
<MaskTestName> String with the name of the mask test

Return values:

<StatusBitNumber> Bit number, see "STATus:QUEStionable:MASK register" on page 1252.

Range: 0 to 7
Increment: 1
*RST: 0

Example:

```
:MTESt:SBITnumber? 'MT1'
<--0
:MTESt:SBITnumber? 'MT2'
<--1
:MTESt:SBITnumber? 'MT3'
<--2
```

Usage: Query only
Asynchronous command

24.27 Remote trace

The commands in this section configure tracing of the remote control interface and of events. They also configure the display of the SCPI remote trace.

Before you start tracing, configure all settings as desired. Modifying settings while tracing is active may result in loss of already traced data. Useful exception: Selecting a new target file while tracing is allowed. For start mode `EXPLicit` a restart of the instrument resets the settings to the documented default values.

If you want to start tracing already during startup of the instrument, configure all settings (including start mode `AUTO`). Then restart your instrument. Tracing will be started automatically during the restart, using the already configured settings.

If you use an XML file as trace file, ensure that tracing is stopped properly. If tracing is aborted instead of stopped, for example by shutting down the instrument for stop mode `EXPLicit`, the XML file will be invalid, because some tags are not closed.

When the maximum file size is reached (except for stop mode `BUFFerfull`) or if tracing is started with an already existing trace file, a backup of the trace file is created and the file itself is reset and overwritten. When the file is full for the second time or when tracing is started the next time, the first backup file is lost because it is overwritten by the next backup. In order to prevent loss of data, set a sufficient file size, select an appropriate stop mode and archive/copy completed trace files if you want to keep them.

24.27.1 Standard commands

TRACe:REMote:MODE:FILE:NAME 2579
TRACe:REMote:MODE:FILE:FORMat 2579
TRACe:REMote:MODE:FILE:SIZE 2579
TRACe:REMote:MODE:FILE:STARtmode 2579

TRACe:REMote:MODE:FILE:STOPmode....2579
TRACe:REMote:MODE:FILE:ENABle....2580
TRACe:REMote:MODE:FILE:FILTer....2580

TRACe:REMote:MODE:FILE:NAME <FilePath>

Sets the directory and file name where the remote trace file is stored.

Parameters:

<FilePath> String parameter

TRACe:REMote:MODE:FILE:FORMat <Format>

Sets the file format of the remote trace file.

Parameters:

<Format> ASCii | XML

TRACe:REMote:MODE:FILE:SIZE <FileSize>

Sets the maximum size of the remote trace file.

Parameters:

<FileSize> Integer value

TRACe:REMote:MODE:FILE:STARtmode <StartMode>

Defines how the remote trace is started.

Parameters:

<StartMode> AUTO | EXPLicit

AUTO
Starts the remote trace immediately.

EXPLicit
Starts remote trace with `TRACe:REMote:MODE:FILE:ENABle ON`

TRACe:REMote:MODE:FILE:STOPmode <StopMode>

Defines when the remote trace is stopped.

Parameters:

<StopMode> AUTO | EXPLicit | ERRor | BUFFerfull

AUTO
Ends the remote trace on device shutdown.

EXPLicit
Ends remote trace with `TRACe:REMote:MODE:FILE:ENABle OFF`

ERRor
Ends remote trace when a SCPI error occurs.

BUFFerfull
Ends remote trace when the maximum file size is reached.

TRACe:REMote:MODE:FILE:ENABle <Enable>

Enables and disables the remote trace to file.

Parameters:

<Enable> ON | OFF

TRACe:REMote:MODE:FILE:FILTer <Input>, <Output>, <Error>, <Trigger>, <DeviceClear>, <StatusRegister>, <Connection>, <RemoteLocalEvents>, <Locking>

Defines the content of the remote trace file.

Parameters:

<Input> ON | OFF
Input data

<Output> ON | OFF
Output data

<Error> ON | OFF
New SCPI error queue entries

<Trigger> ON | OFF
Trigger events

<DeviceClear> ON | OFF
Device clear events

<StatusRegister> ON | OFF
Status register conditions

<Connection> ON | OFF
Open/close connection events

<RemoteLocalEvents> ON | OFF
Local/remote transition events

<Locking> ON | OFF
Remote locking events

24.27.2 Diagnostic remote trace commands

TRACe:REMote:MODE:FILE:DEXecution:DURation....2581
TRACe:REMote:MODE:FILE:RPC....2581
TRACe:REMote:MODE:FILE:PARSer....2581
TRACe:REMote:MODE:FILE:FUNCtions....2581

TRACe:REMote:MODE:FILE:DEXecution:DURation <Enable>

Traces the device execution time of a command

Parameters:
<Enable> ON | OFF

TRACe:REMote:MODE:FILE:RPC <Enable>

Enables and disables output of rpc calls to remote trace.

Parameters:
<Enable> ON | OFF

TRACe:REMote:MODE:FILE:PARSer <Enable>

Enables and disables output of parser transitions to remote trace.

Parameters:
<Enable> ON | OFF

TRACe:REMote:MODE:FILE:FUNCtions <Enable>

Enables and disables output of function names to remote trace.

Parameters:
<Enable> ON | OFF

24.28 Deprecated commands

The following commands are provided for compatibility to previous oscilloscope versions only. For new remote control programs, use the specified alternative commands.

24.28.1 Base instrument

Legacy command	Replaced by	Comment
EXPort:RESult:SELect EXPort:RESult:NUMeric EXPort:RESult:NAME EXPort:RESult:SAVE		Functionality is not supported. Removed since FW 5.20
TRIGger<m>:OUT:STATe	TRIGger<m>:OUT:ACTion on page 1433	Replaced since FW 5.00
TRIGger<m>:SEQuence:MODE	TRIGger<m>:SEQuence:TYPE on page 1427	Replaced since FW 5.10

24.28.2 Options

Legacy command	Replaced by	Comment
I2C		
TRIGger<m>:I2C:MODE	TRIGger<m>:I2C:TYPE on page 1679	Replaced since FW 5.20
SPI		
TRIGger<m>:SPI:MODE SEARch:TRIGger:SPI:MODE	TRIGger<m>:SPI:TYPE on page 1713 SEARch:TRIGger:SPI:TYPE on page 1721	Replaced since FW 5.20
Ethernet		
SEARch:TRIGger:ETHernet:ERRor:SELect SEARch:TRIGger:ETHernet:FRAMe:SELect	SEARch:TRIGger:ETHernet:TYPE on page 2030	Replaced since FW 5.20
Advanced Jitter Analysis		
ADVJitter<m>::RESult:TYPBathtub ADVJitter<m>::RESult:TYPSynteye		Functionality is not supported.
ADVJitter<m>:SIGNal:UIOFfset	ADVJitter<m>:SIGNal:SAMTime on page 2489	Replaced since FW 5.10

List of commands

*CAL?....1288
*CLS....1288
*ESE....1288
*ESR?....1288
*IDN?....1288
*IST?....1289
*OPC....1289
*OPT?....1289
*PCB....1289
*PRE....1290
*PSC....1290
*RCL....1290
*RST....1290
*SAV....1290
*SRE....1291
*STB?....1291
*TRG....1291
*TST?....1291
*WAI....1292
ACQuire:ARESet:COUNt....1335
ACQuire:ARESet:COUNt....2538
ACQuire:ARESet:IMMediate....1335
ACQuire:ARESet:MODE....1335
ACQuire:ARESet:MODE....2537
ACQuire:ARESet:TIME....1335
ACQuire:ARESet:TIME....2538
ACQuire:AVAilable?....1461
ACQuire:CDTA....1332
ACQuire:COUNt....1334
ACQuire:CURRent?....1437
ACQuire:DRESolution?....2430
ACQuire:INTerpolate....1332
ACQuire:MUWaveform....1333
ACQuire:POINts:AADJust....1329
ACQuire:POINts:ARATe?....1329
ACQuire:POINts:AUTO....1328
ACQuire:POINts:DVALue?....2431
ACQuire:POINts:MAXimum....1329
ACQuire:POINts:ZVALue?....1377
ACQuire:POINts[:VALue]....1330
ACQuire:RESolution....1330
ACQuire:SEGMented:AUToreplay....1336
ACQuire:SEGMented:MAX....1336
ACQuire:SEGMented:STATe....1336
ACQuire:SRATe....1329
ACQuire:SRReal....1330
ACQuire:ZRESolution?....1378

ADVJitter<m>:DCOMposition:CLEBasicmeas....2499
ADVJitter<m>:DCOMposition:CLEJitcomp....2499
ADVJitter<m>:DCOMposition:CLENoisecomp....2499
ADVJitter<m>:DCOMposition:COMPonents<n>:ENABle....2495
ADVJitter<m>:DCOMposition:ENOBu....2509
ADVJitter<m>:DCOMposition:ENP....2509
ADVJitter<m>:DCOMposition:JINTerleaved....2507
ADVJitter<m>:DCOMposition:MRESplength?....2508
ADVJitter<m>:DCOMposition:NHORizontal....2509
ADVJitter<m>:DCOMposition:NVERtical....2509
ADVJitter<m>:DCOMposition:POSStep....2508
ADVJitter<m>:DCOMposition:RESPlength....2508
ADVJitter<m>:DCOMposition:RLEStimate....2508
ADVJitter<m>:DCOMposition:SELBasicmeas....2499
ADVJitter<m>:DCOMposition:SELJitcomp....2499
ADVJitter<m>:DCOMposition:SELNoisecomp....2499
ADVJitter<m>:DCOMposition:SPJLabels....2507
ADVJitter<m>:DCOMposition:SPJTable....2506
ADVJitter<m>:DCOMposition:SPNLabels....2507
ADVJitter<m>:DCOMposition:TARBer....2500
ADVJitter<m>:RESult:BATHtub....2501
ADVJitter<m>:RESult:COMPonents<n>:HISTogram....2495
ADVJitter<m>:RESult:COMPonents<n>:OFFSet....2506
ADVJitter<m>:RESult:COMPonents<n>:SCALe....2505
ADVJitter<m>:RESult:COMPonents<n>:SPECtrum....2496
ADVJitter<m>:RESult:COMPonents<n>:STATistics:MAXPeakpeak?....2496
ADVJitter<m>:RESult:COMPonents<n>:STATistics:MAXStddev?....2496
ADVJitter<m>:RESult:COMPonents<n>:STATistics:MINPeakpeak?....2496
ADVJitter<m>:RESult:COMPonents<n>:STATistics:PEAKpeak?....2496
ADVJitter<m>:RESult:COMPonents<n>:STATistics:STDDev?....2497
ADVJitter<m>:RESult:COMPonents<n>:STATistics:WFMCount?....2497
ADVJitter<m>:RESult:COMPonents<n>:STATistics[:MINStddev]?....2496
ADVJitter<m>:RESult:COMPonents<n>:TRACk....2496
ADVJitter<m>:RESult:ENABle....2494
ADVJitter<m>:RESult:LEVels....2500
ADVJitter<m>:RESult:NBAThtub....2502
ADVJitter<m>:RESult:OBUJ....2501
ADVJitter<m>:RESult:OBUN....2502
ADVJitter<m>:RESult:PERiodic<n>:COUNt?....2497
ADVJitter<m>:RESult:PERiodic<n>:DIRection?....2497
ADVJitter<m>:RESult:PERiodic<n>:VALue?....2498
ADVJitter<m>:RESult:PERiodic<n>[:FREQuency]?....2498
ADVJitter<m>:RESult:PHORizontal....2502
ADVJitter<m>:RESult:PJ....2501
ADVJitter<m>:RESult:PN....2502
ADVJitter<m>:RESult:PVERtical....2503
ADVJitter<m>:RESult:STATistics....2495
ADVJitter<m>:RESult:STEPresponse....2495
ADVJitter<m>:RESult:SYNTheticeye....2503
ADVJitter<m>:RESult:TRANsitions....2500

ADVJitter<m>:RESult:TYPBathtub....2501
ADVJitter<m>:RESult:TYPSynteye....2503
ADVJitter<m>:RESult:UNIT....2496
ADVJitter<m>:RESult:VERTical:BHContinuous....2504
ADVJitter<m>:RESult:VERTical:BMTYpe....2504
ADVJitter<m>:RESult:VERTical:CONTinuous....2504
ADVJitter<m>:RESult:VERTical:HBINs....2504
ADVJitter<m>:RESult:VERTical:JSContinuous....2505
ADVJitter<m>:RESult:VERTical:NHContinuous....2505
ADVJitter<m>:RESult:VERTical:NSContinuous....2505
ADVJitter<m>:SIGNal:BITRate....2486
ADVJitter<m>:SIGNal:CDRMode....2486
ADVJitter<m>:SIGNal:CFEStimation....2488
ADVJitter<m>:SIGNal:CLEStimation....2487
ADVJitter<m>:SIGNal:CLFFactor....2487
ADVJitter<m>:SIGNal:CLFRequency....2488
ADVJitter<m>:SIGNal:CLKSource....2487
ADVJitter<m>:SIGNal:CLOFfset....2488
ADVJitter<m>:SIGNal:DATRate?....2488
ADVJitter<m>:SIGNal:EDGE....2487
ADVJitter<m>:SIGNal:GATE:COUPling....2489
ADVJitter<m>:SIGNal:GATE:CURSor....2489
ADVJitter<m>:SIGNal:GATE:RANGe:ABSolute:STARt....2489
ADVJitter<m>:SIGNal:GATE:RANGe:ABSolute:STOP....2490
ADVJitter<m>:SIGNal:GATE:RANGe:MODE....2490
ADVJitter<m>:SIGNal:GATE:RANGe:RELative:STARt....2490
ADVJitter<m>:SIGNal:GATE:RANGe:RELative:STOP....2490
ADVJitter<m>:SIGNal:GATE:STATe....2491
ADVJitter<m>:SIGNal:GATE:ZDIagram....2491
ADVJitter<m>:SIGNal:QUICkmeas....2486
ADVJitter<m>:SIGNal:REFSource....2486
ADVJitter<m>:SIGNal:SAMTime....2489
ADVJitter<m>:SIGNal:SERStandard....2491
ADVJitter<m>:SIGNal:SOURce....2486
ADVJitter<m>:SIGNal:TYPE....2485
AUToscale....1328
BUS<m>:ARINc:BRMode....1983
BUS<m>:ARINc:BRValue....1983
BUS<m>:ARINc:MAXGap:BITS....1984
BUS<m>:ARINc:MAXGap:SELect....1984
BUS<m>:ARINc:MINGap:BITS....1984
BUS<m>:ARINc:MINGap:SELect....1984
BUS<m>:ARINc:POLarity....1985
BUS<m>:ARINc:PRESet....1985
BUS<m>:ARINc:SOURce....1983
BUS<m>:ARINc:THReshold:HIGH....1985
BUS<m>:ARINc:THReshold:LOW....1985
BUS<m>:ARINc:WCOunt?....1990
BUS<m>:ARINc:WORD<n>:DATA?....1990
BUS<m>:ARINc:WORD<n>:LABel?....1990

BUS<m>:ARINc:WORD<n>:PATTern?....1991
BUS<m>:ARINc:WORD<n>:SDI?....1991
BUS<m>:ARINc:WORD<n>:SSM?....1991
BUS<m>:ARINc:WORD<n>:STARt?....1992
BUS<m>:ARINc:WORD<n>:STATe?....1992
BUS<m>:ARINc:WORD<n>:STOP?....1992
BUS<m>:ARINc:WORD<n>:SYMBol?....1993
BUS<m>:CAN:BITRate....1797
BUS<m>:CAN:DATA:SOURce....1795
BUS<m>:CAN:DATA:THReshold....1796
BUS<m>:CAN:FCOunt?....1808
BUS<m>:CAN:FDATa:DBITrate....1798
BUS<m>:CAN:FDATa:ENABle....1798
BUS<m>:CAN:FDATa:FRAMe<n>:SCValue?....1816
BUS<m>:CAN:FDATa:FRAMe<n>:STANdard?....1809
BUS<m>:CAN:FDATa:JWIDth....1800
BUS<m>:CAN:FDATa:PSTandard....1796
BUS<m>:CAN:FDATa:SAMPlepoint....1798
BUS<m>:CAN:FDATa:T1Segment....1799
BUS<m>:CAN:FDATa:T2Segment....1799
BUS<m>:CAN:FRAMe<n>:ACKState?....1812
BUS<m>:CAN:FRAMe<n>:ACKValue?....1812
BUS<m>:CAN:FRAMe<n>:BITRate?....1813
BUS<m>:CAN:FRAMe<n>:BSEPosition?....1815
BUS<m>:CAN:FRAMe<n>:BYTE<o>:STATe?....1816
BUS<m>:CAN:FRAMe<n>:BYTE<o>:VALue?....1816
BUS<m>:CAN:FRAMe<n>:CSSTate?....1812
BUS<m>:CAN:FRAMe<n>:CSValue?....1813
BUS<m>:CAN:FRAMe<n>:DATA?....1811
BUS<m>:CAN:FRAMe<n>:DLCState?....1812
BUS<m>:CAN:FRAMe<n>:DLCValue?....1813
BUS<m>:CAN:FRAMe<n>:FERCause?....1815
BUS<m>:CAN:FRAMe<n>:IDSTate?....1812
BUS<m>:CAN:FRAMe<n>:IDTYpe?....1814
BUS<m>:CAN:FRAMe<n>:IDValue?....1814
BUS<m>:CAN:FRAMe<n>:NDBYtes?....1810
BUS<m>:CAN:FRAMe<n>:SDATa?....1833
BUS<m>:CAN:FRAMe<n>:SDEXport?....1815
BUS<m>:CAN:FRAMe<n>:STARt?....1810
BUS<m>:CAN:FRAMe<n>:STATus?....1809
BUS<m>:CAN:FRAMe<n>:STOP?....1810
BUS<m>:CAN:FRAMe<n>:SYMBol?....1811
BUS<m>:CAN:FRAMe<n>:TYPE?....1811
BUS<m>:CAN:JWIDth....1800
BUS<m>:CAN:SAMPlepoint....1798
BUS<m>:CAN:T1Segment....1799
BUS<m>:CAN:T2Segment....1799
BUS<m>:CAN:TECHnology....1797
BUS<m>:CAN:TYPE....1796
BUS<m>:CDR:BITRate....2349

BUS<m>:CDR:PLL:BWIDth....2349
BUS<m>:CDR:PLL:DAMPing....2350
BUS<m>:CDR:PLL:ORDer....2350
BUS<m>:CDR:PLL:RELBwidth....2350
BUS<m>:CDR:RESults....2350
BUS<m>:CDR:SYNC....2351
BUS<m>:CMSB:ADDFrame....1763
BUS<m>:CMSB:BITRate:ENABle....1762
BUS<m>:CMSB:BITRate:VALue....1762
BUS<m>:CMSB:CLR....1763
BUS<m>:CMSB:CODing....1753
BUS<m>:CMSB:EXRBits....1766
BUS<m>:CMSB:FCOunt?....1763
BUS<m>:CMSB:FILTer:ENABle....1774
BUS<m>:CMSB:FILTer:ERENable....1774
BUS<m>:CMSB:FILTer:ERRor<n>:ENABle....1774
BUS<m>:CMSB:FILTer:FIENable....1775
BUS<m>:CMSB:FILTer:FRAMe<n>:ENABle....1775
BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:BIT....1775
BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:DMAX....1776
BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:DMIN....1776
BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:DOPerator....1776
BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:ENABle....1777
BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:IMAX....1777
BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:IMIN....1777
BUS<m>:CMSB:FILTer:FRAMe<n>:FLD<o>:IOPerator....1777
BUS<m>:CMSB:FILTer:FRENable....1775
BUS<m>:CMSB:FRAMe<n>:APPend....1764
BUS<m>:CMSB:FRAMe<n>:CCOunt?....1764
BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITCount....1764
BUS<m>:CMSB:FRAMe<n>:CELL<o>:BITorder....1766
BUS<m>:CMSB:FRAMe<n>:CELL<o>:CLMN....1766
BUS<m>:CMSB:FRAMe<n>:CELL<o>:CONDition....1765
BUS<m>:CMSB:FRAMe<n>:CELL<o>:CRGB....1766
BUS<m>:CMSB:FRAMe<n>:CELL<o>:FORMat....1765
BUS<m>:CMSB:FRAMe<n>:CELL<o>:NAME....1764
BUS<m>:CMSB:FRAMe<n>:TYPE....1763
BUS<m>:CMSB:GAPTime:ENABle....1762
BUS<m>:CMSB:GAPTime:VALue....1762
BUS<m>:CMSB:LOAD....1767
BUS<m>:CMSB:MANChester:CPHase....1756
BUS<m>:CMSB:MANChester:DATA....1754
BUS<m>:CMSB:MANChester:POLarity....1754
BUS<m>:CMSB:MANChester:THReshold:COUPling....1755
BUS<m>:CMSB:MANChester:THReshold:HIGH....1754
BUS<m>:CMSB:MANChester:THReshold:LOW....1754
BUS<m>:CMSB:MANChester:THReshold:PRESet....1755
BUS<m>:CMSB:NRZ:CLCK....1756
BUS<m>:CMSB:NRZ:CPHase....1758
BUS<m>:CMSB:NRZ:CPOLarity....1757

BUS<m>:CMSB:NRZ:DATA.... 1756
BUS<m>:CMSB:NRZ:ENAPolarity.... 1759
BUS<m>:CMSB:NRZ:ENBLe.... 1758
BUS<m>:CMSB:NRZ:HYSTeresis:CLCK.... 1759
BUS<m>:CMSB:NRZ:HYSTeresis:DATA.... 1760
BUS<m>:CMSB:NRZ:HYSTeresis:ENBLe.... 1760
BUS<m>:CMSB:NRZ:IDLPolarity.... 1757
BUS<m>:CMSB:NRZ:POLarity.... 1759
BUS<m>:CMSB:NRZ:THReshold:CLCK.... 1759
BUS<m>:CMSB:NRZ:THReshold:COUPling.... 1761
BUS<m>:CMSB:NRZ:THReshold:DATA.... 1760
BUS<m>:CMSB:NRZ:THReshold:ENBLe.... 1760
BUS<m>:CMSB:NRZ:THReshold:PRESet.... 1761
BUS<m>:CMSB:RCOunt?.... 1779
BUS<m>:CMSB:RESult<n>:CCOunt?.... 1782
BUS<m>:CMSB:RESult<n>:CELL<o>:NAME?.... 1782
BUS<m>:CMSB:RESult<n>:CELL<o>:STARt?.... 1782
BUS<m>:CMSB:RESult<n>:CELL<o>:STATe?.... 1783
BUS<m>:CMSB:RESult<n>:CELL<o>:STOP?.... 1783
BUS<m>:CMSB:RESult<n>:CELL<o>:VALue?.... 1784
BUS<m>:CMSB:RESult<n>:CONE?.... 1781
BUS<m>:CMSB:RESult<n>:CTHRee?.... 1781
BUS<m>:CMSB:RESult<n>:CTWO?.... 1781
BUS<m>:CMSB:RESult<n>:STARt?.... 1780
BUS<m>:CMSB:RESult<n>:STATe?.... 1779
BUS<m>:CMSB:RESult<n>:STOP?.... 1780
BUS<m>:CMSB:RESult<n>:TYPE?.... 1781
BUS<m>:CMSB:SAVE.... 1767
BUS<m>:CTLE:DCGain.... 2190
BUS<m>:CTLE:ENBLe.... 2190
BUS<m>:CTLE:P1FReq.... 2190
BUS<m>:CTLE:P2FReq.... 2191
BUS<m>:CTLE:Z1FReq.... 2191
BUS<m>:CXPI:BITRate:ENABle.... 1930
BUS<m>:CXPI:BITRate:VALue.... 1930
BUS<m>:CXPI:DORD.... 1931
BUS<m>:CXPI:HYSTeresis.... 1931
BUS<m>:CXPI:IBS:MAX.... 1932
BUS<m>:CXPI:IBS:MIN.... 1932
BUS<m>:CXPI:IFS:MAX.... 1932
BUS<m>:CXPI:IFS:MIN.... 1933
BUS<m>:CXPI:POLarity.... 1931
BUS<m>:CXPI:RESult:BITRate?.... 1930
BUS<m>:CXPI:RESult:FCOunt?.... 1941
BUS<m>:CXPI:RESult:FRAMe<n>:DATA?.... 1942
BUS<m>:CXPI:RESult:FRAMe<n>:DLCV?.... 1942
BUS<m>:CXPI:RESult:FRAMe<n>:STARt?.... 1942
BUS<m>:CXPI:RESult:FRAMe<n>:STATe?.... 1943
BUS<m>:CXPI:RESult:FRAMe<n>:STOP?.... 1943
BUS<m>:CXPI:RESult:FRAMe<n>:TYPE?.... 1943

BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:STATus? 1944
BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:TYPE? 1944
BUS<m>:CXPI:RESult:FRAMe<n>:WORD<o>:VALue? 1944
BUS<m>:CXPI:SDATa 1932
BUS<m>:CXPI:THReshold 1931
BUS<m>:DDR:BITRate 2389
BUS<m>:DDR:DATA:HYSTeresis 2387
BUS<m>:DDR:DATA:SOURce 2388
BUS<m>:DDR:DATA:THReshold 2388
BUS<m>:DDR:RESult:FCOunt? 2390
BUS<m>:DDR:RESult:FRAMe<n>:FLD<o>:FVAL? 2390
BUS<m>:DDR:RESult:FRAMe<n>:FLD<o>:STARt? 2390
BUS<m>:DDR:RESult:FRAMe<n>:FLD<o>:STATus? 2391
BUS<m>:DDR:RESult:FRAMe<n>:FLD<o>:STOP? 2391
BUS<m>:DDR:RESult:FRAMe<n>:FLD<o>:VAL? 2391
BUS<m>:DDR:RESult:FRAMe<n>:NWRDs? 2392
BUS<m>:DDR:RESult:FRAMe<n>:STARt? 2392
BUS<m>:DDR:RESult:FRAMe<n>:STATe? 2392
BUS<m>:DDR:RESult:FRAMe<n>:STOP? 2393
BUS<m>:DDR:RESult:FRAMe<n>:TYPE? 2393
BUS<m>:DDR:STANdard 2387
BUS<m>:DDR:STRBe:HYSTeresis 2388
BUS<m>:DDR:STRBe:SOURce 2388
BUS<m>:DDR:STRBe:THReshold 2389
BUS<m>:DPHY:CONFig 2243
BUS<m>:DPHY:CP:HSPeed:HYSTeresis 2247
BUS<m>:DPHY:CP:HSPeed:THReshold 2247
BUS<m>:DPHY:CP:PROBe 2244
BUS<m>:DPHY:CP:SOURce 2244
BUS<m>:DPHY:DNZero:LPOWer:THLower 2246
BUS<m>:DPHY:DNZero:LPOWer:THUPper 2246
BUS<m>:DPHY:DNZero:PROBe 2247
BUS<m>:DPHY:DNZero:SOURce 2246
BUS<m>:DPHY:DPONe:HSPeed:HYSTeresis 2247
BUS<m>:DPHY:DPONe:HSPeed:THReshold 2247
BUS<m>:DPHY:DPONe:PROBe 2247
BUS<m>:DPHY:DPONe:SOURce 2246
BUS<m>:DPHY:DPTHree:HSPeed:HYSTeresis 2247
BUS<m>:DPHY:DPTHree:HSPeed:THReshold 2247
BUS<m>:DPHY:DPTHree:PROBe 2247
BUS<m>:DPHY:DPTHree:SOURce 2246
BUS<m>:DPHY:DPTWo:HSPeed:HYSTeresis 2247
BUS<m>:DPHY:DPTWo:HSPeed:THReshold 2247
BUS<m>:DPHY:DPTWo:PROBe 2247
BUS<m>:DPHY:DPTWo:SOURce 2246
BUS<m>:DPHY:DPZero:HSPeed:HYSTeresis 2247
BUS<m>:DPHY:DPZero:HSPeed:THReshold 2247
BUS<m>:DPHY:DPZero:LPOWer:THLower 2246
BUS<m>:DPHY:DPZero:LPOWer:THUPper 2246
BUS<m>:DPHY:DPZero:PROBe 2247

BUS<m>:DPHY:DPZero:SOURce....2246
BUS<m>:DPHY:DRATe....2244
BUS<m>:DPHY:DSPData....2245
BUS<m>:DPHY:RESult:FCOunt?....2255
BUS<m>:DPHY:RESult:FRAMe<n>:CS?....2255
BUS<m>:DPHY:RESult:FRAMe<n>:DATA?....2256
BUS<m>:DPHY:RESult:FRAMe<n>:DTName?....2256
BUS<m>:DPHY:RESult:FRAMe<n>:DTYPe?....2257
BUS<m>:DPHY:RESult:FRAMe<n>:ECC?....2257
BUS<m>:DPHY:RESult:FRAMe<n>:NUMPackets?....2260
BUS<m>:DPHY:RESult:FRAMe<n>:PACKet<o>:IDX?....2257
BUS<m>:DPHY:RESult:FRAMe<n>:PACKet<o>:VALue?....2258
BUS<m>:DPHY:RESult:FRAMe<n>:STARt?....2258
BUS<m>:DPHY:RESult:FRAMe<n>:STATe?....2258
BUS<m>:DPHY:RESult:FRAMe<n>:STOP?....2259
BUS<m>:DPHY:RESult:FRAMe<n>:TYPE?....2259
BUS<m>:DPHY:RESult:FRAMe<n>:VCHannel?....2260
BUS<m>:DPHY:SMLanes....2248
BUS<m>:DPHY:THCoupling....2245
BUS<m>:DPHY:THPReset....2245
BUS<m>:DPHY:VARiant....2245
BUS<m>:EBTB:BITDetermi....2336
BUS<m>:EBTB:BITRate....2332
BUS<m>:EBTB:DIFFerential:SOURce....2332
BUS<m>:EBTB:DIFFerential:THRHigh....2332
BUS<m>:EBTB:DIFFerential:THRLow....2333
BUS<m>:EBTB:DISF....2333
BUS<m>:EBTB:DMINus:SOURce....2333
BUS<m>:EBTB:DMINus:THReshold....2333
BUS<m>:EBTB:DPLus:SOURce....2334
BUS<m>:EBTB:DPLus:THReshold....2334
BUS<m>:EBTB:EBTRate?....2332
BUS<m>:EBTB:FAUToscale....2336
BUS<m>:EBTB:FCSY....2334
BUS<m>:EBTB:SCOunt?....2334
BUS<m>:EBTB:SCSY....2335
BUS<m>:EBTB:SYMBol<n>:BYTE?....2340
BUS<m>:EBTB:SYMBol<n>:DATA?....2339
BUS<m>:EBTB:SYMBol<n>:KDCode?....2341
BUS<m>:EBTB:SYMBol<n>:STARt?....2339
BUS<m>:EBTB:SYMBol<n>:STATus?....2340
BUS<m>:EBTB:SYMBol<n>:STOP?....2340
BUS<m>:EBTB:SYMBol<n>:TENBit?....2341
BUS<m>:EBTB:SYNC....2335
BUS<m>:EBTB:TYPE....2335
BUS<m>:EBTB:USCS....2335
BUS<m>:ETHernet:BITRate....2019
BUS<m>:ETHernet:POLarity....2017
BUS<m>:ETHernet:PRESet....2018
BUS<m>:ETHernet:SOURce....2017

BUS<m>:ETHernet:THReshold:HIGH....2018
BUS<m>:ETHernet:THReshold:LOW....2018
BUS<m>:ETHernet:VARiant....2017
BUS<m>:ETHernet:WCOunt?....2025
BUS<m>:ETHernet:WORD<n>:BITRate?....2029
BUS<m>:ETHernet:WORD<n>:BYTE<o>:VALue?....2029
BUS<m>:ETHernet:WORD<n>:CRC?....2028
BUS<m>:ETHernet:WORD<n>:DATA?....2028
BUS<m>:ETHernet:WORD<n>:DESTaddress?....2027
BUS<m>:ETHernet:WORD<n>:DSYMbol?....2028
BUS<m>:ETHernet:WORD<n>:FTYPe?....2025
BUS<m>:ETHernet:WORD<n>:NUMWords?....2030
BUS<m>:ETHernet:WORD<n>:SRCaddress?....2027
BUS<m>:ETHernet:WORD<n>:SSYMbol?....2029
BUS<m>:ETHernet:WORD<n>:STARt?....2026
BUS<m>:ETHernet:WORD<n>:STATe?....2026
BUS<m>:ETHernet:WORD<n>:STOP?....2026
BUS<m>:ETHernet:WORD<n>:TYPE?....2027
BUS<m>:EXPResult:DETail....1673
BUS<m>:EXPResult:SAVE....1673
BUS<m>:EXPResult:TIME....1673
BUS<m>:FAUToset....1671
BUS<m>:FLXRay:BITRate....1869
BUS<m>:FLXRay:CHTYpe....1869
BUS<m>:FLXRay:FCOunt?....1878
BUS<m>:FLXRay:FRAMe<n>:ADID?....1881
BUS<m>:FLXRay:FRAMe<n>:CSSTate?....1882
BUS<m>:FLXRay:FRAMe<n>:CSValue?....1883
BUS<m>:FLXRay:FRAMe<n>:CYCount?....1882
BUS<m>:FLXRay:FRAMe<n>:DATA?....1880
BUS<m>:FLXRay:FRAMe<n>:FCSTate?....1883
BUS<m>:FLXRay:FRAMe<n>:FCValue?....1883
BUS<m>:FLXRay:FRAMe<n>:FLAGs?....1881
BUS<m>:FLXRay:FRAMe<n>:PAYLength?....1882
BUS<m>:FLXRay:FRAMe<n>:STARt?....1879
BUS<m>:FLXRay:FRAMe<n>:STATus?....1879
BUS<m>:FLXRay:FRAMe<n>:STOP?....1879
BUS<m>:FLXRay:FRAMe<n>:SYMBol?....1880
BUS<m>:FLXRay:FRAMe<n>:TYPE?....1880
BUS<m>:FLXRay:POLarity....1869
BUS<m>:FLXRay:PRDiff....1868
BUS<m>:FLXRay:PRLogic....1868
BUS<m>:FLXRay:PRSingle....1867
BUS<m>:FLXRay:SEHB....1869
BUS<m>:FLXRay:SOURce<n>....1866
BUS<m>:FLXRay:SRCType....1865
BUS<m>:FLXRay:THData....1867
BUS<m>:FLXRay:THENable....1867
BUS<m>:FLXRay:THReshold<n>....1867
BUS<m>:FORMat....1672

BUS<m>:HBTO:ATTN ... 2087
BUS<m>:HBTO:FDAM ... 2087
BUS<m>:HBTO:FDAP ... 2087
BUS<m>:HBTO:FDIF ... 2087
BUS<m>:HBTO:FTYP ... 2088
BUS<m>:HBTO:MODE ... 2089
BUS<m>:HBTO:POLarity ... 2089
BUS<m>:HBTO:RDAM ... 2088
BUS<m>:HBTO:RDAP ... 2088
BUS<m>:HBTO:RDIF ... 2088
BUS<m>:HBTO:RESult:FCOunt? ... 2098
BUS<m>:HBTO:RESult:FRAMe<n>:BITRate? ... 2102
BUS<m>:HBTO:RESult:FRAMe<n>:CRC? ... 2101
BUS<m>:HBTO:RESult:FRAMe<n>:DATA? ... 2100
BUS<m>:HBTO:RESult:FRAMe<n>:DESTaddress? ... 2100
BUS<m>:HBTO:RESult:FRAMe<n>:DSYMbol? ... 2101
BUS<m>:HBTO:RESult:FRAMe<n>:FTYPe? ... 2098
BUS<m>:HBTO:RESult:FRAMe<n>:NUMWords? ... 2101
BUS<m>:HBTO:RESult:FRAMe<n>:SRCaddress? ... 2100
BUS<m>:HBTO:RESult:FRAMe<n>:SSYMbol? ... 2101
BUS<m>:HBTO:RESult:FRAMe<n>:STARt? ... 2099
BUS<m>:HBTO:RESult:FRAMe<n>:STATe? ... 2098
BUS<m>:HBTO:RESult:FRAMe<n>:STOP? ... 2099
BUS<m>:HBTO:RESult:FRAMe<n>:TYPE? ... 2100
BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:TYPE? ... 2102
BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:VALue? ... 2102
BUS<m>:HBTO:RESult:FRAMe<n>:WORD<o>:VSTR? ... 2103
BUS<m>:HBTO:RTYP ... 2089
BUS<m>:HBTO:SYMRate ... 2089
BUS<m>:HBTO:THReshold ... 2089
BUS<m>:I2C:FCOunt? ... 1685
BUS<m>:I2C:FRAMe<n>:AACCess? ... 1685
BUS<m>:I2C:FRAMe<n>:ACCess? ... 1686
BUS<m>:I2C:FRAMe<n>:ACOMplete? ... 1686
BUS<m>:I2C:FRAMe<n>:ADBStart? ... 1686
BUS<m>:I2C:FRAMe<n>:ADDRess? ... 1687
BUS<m>:I2C:FRAMe<n>:ADEVice? ... 1687
BUS<m>:I2C:FRAMe<n>:AMODe? ... 1687
BUS<m>:I2C:FRAMe<n>:ASTart? ... 1688
BUS<m>:I2C:FRAMe<n>:BCOunt? ... 1690
BUS<m>:I2C:FRAMe<n>:BITRate? ... 1688
BUS<m>:I2C:FRAMe<n>:BYTE<o>:ACCess? ... 1691
BUS<m>:I2C:FRAMe<n>:BYTE<o>:ACKStart? ... 1691
BUS<m>:I2C:FRAMe<n>:BYTE<o>:COMPlete? ... 1691
BUS<m>:I2C:FRAMe<n>:BYTE<o>:STARt? ... 1692
BUS<m>:I2C:FRAMe<n>:BYTE<o>:VALue? ... 1692
BUS<m>:I2C:FRAMe<n>:DATA? ... 1685
BUS<m>:I2C:FRAMe<n>:RWBStart? ... 1688
BUS<m>:I2C:FRAMe<n>:STARt? ... 1689
BUS<m>:I2C:FRAMe<n>:STATus? ... 1689

BUS<m>:I2C:FRAMe<n>:STOP?....1690
BUS<m>:I2C:FRAMe<n>:SYMBol?....1690
BUS<m>:I2C:RWBit....1678
BUS<m>:I2C:SCL:SOURce....1676
BUS<m>:I2C:SCL:THReshold....1677
BUS<m>:I2C:SDA:SOURce....1676
BUS<m>:I2C:SDA:THReshold....1677
BUS<m>:I2C:TECHnology....1677
BUS<m>:I2S:AVARiant....1737
BUS<m>:I2S:BORDer....1742
BUS<m>:I2S:CHANnel:LENGth....1743
BUS<m>:I2S:CHANnel:OFFSet....1742
BUS<m>:I2S:CHANnel:ORDer....1741
BUS<m>:I2S:CHANnel:TDMCount....1742
BUS<m>:I2S:CLOCk:POLarity....1738
BUS<m>:I2S:CLOCk:SOURce....1738
BUS<m>:I2S:CLOCk:THReshold....1740
BUS<m>:I2S:DATA:POLarity....1740
BUS<m>:I2S:DATA:SOURce....1739
BUS<m>:I2S:DATA:THReshold....1741
BUS<m>:I2S:FCOunt?....1747
BUS<m>:I2S:FOFFset....1742
BUS<m>:I2S:FRAMe<n>:LEFT:STATe?....1749
BUS<m>:I2S:FRAMe<n>:LEFT:VALue?....1748
BUS<m>:I2S:FRAMe<n>:RIGHt:STATe?....1749
BUS<m>:I2S:FRAMe<n>:RIGHt:VALue?....1748
BUS<m>:I2S:FRAMe<n>:STARt?....1748
BUS<m>:I2S:FRAMe<n>:STATe?....1747
BUS<m>:I2S:FRAMe<n>:STOP?....1748
BUS<m>:I2S:FRAMe<n>:TDM<o>:STATe?....1749
BUS<m>:I2S:FRAMe<n>:TDM<o>:VALue?....1749
BUS<m>:I2S:TCOupling....1740
BUS<m>:I2S:TRACk:LEFT....1750
BUS<m>:I2S:TRACk:RIGHt....1750
BUS<m>:I2S:TRACk:TD1Ch....1750
BUS<m>:I2S:TRACk:TD2Ch....1750
BUS<m>:I2S:TRACk:TD3Ch....1750
BUS<m>:I2S:TRACk:TD4Ch....1750
BUS<m>:I2S:TRACk:TD5Ch....1751
BUS<m>:I2S:TRACk:TD6Ch....1751
BUS<m>:I2S:TRACk:TD7Ch....1751
BUS<m>:I2S:TRACk:TD8Ch....1751
BUS<m>:I2S:WLENgth....1741
BUS<m>:I2S:WSELect:POLarity....1739
BUS<m>:I2S:WSELect:SOURce....1738
BUS<m>:I2S:WSELect:THReshold....1740
BUS<m>:LIN:BITRate....1838
BUS<m>:LIN:DATA:SOURce....1837
BUS<m>:LIN:DATA:THReshold....1837
BUS<m>:LIN:FCOunt?....1845

BUS<m>:LIN:FRAMe<n>:BITRate?.... 1851
BUS<m>:LIN:FRAMe<n>:BYTE<o>:STATe?.... 1851
BUS<m>:LIN:FRAMe<n>:BYTE<o>:VALue?.... 1852
BUS<m>:LIN:FRAMe<n>:CSSTate?.... 1850
BUS<m>:LIN:FRAMe<n>:CSValue?.... 1850
BUS<m>:LIN:FRAMe<n>:DATA?.... 1848
BUS<m>:LIN:FRAMe<n>:IDPValue?.... 1849
BUS<m>:LIN:FRAMe<n>:IDSTate?.... 1848
BUS<m>:LIN:FRAMe<n>:IDValue?.... 1849
BUS<m>:LIN:FRAMe<n>:SDATa?.... 1847
BUS<m>:LIN:FRAMe<n>:SDEXport?.... 1847
BUS<m>:LIN:FRAMe<n>:STARt?.... 1846
BUS<m>:LIN:FRAMe<n>:STATus?.... 1846
BUS<m>:LIN:FRAMe<n>:STOP?.... 1846
BUS<m>:LIN:FRAMe<n>:SYMBol?.... 1847
BUS<m>:LIN:FRAMe<n>:SYSTate?.... 1850
BUS<m>:LIN:FRAMe<n>:VERSion?.... 1848
BUS<m>:LIN:POLarity.... 1838
BUS<m>:LIN:STANdard.... 1839
BUS<m>:LIN:TECHnology.... 1838
BUS<m>:MDIO:CLOCk:SOURce.... 2040
BUS<m>:MDIO:CLOCk:THReshold:HIGH.... 2041
BUS<m>:MDIO:CLOCk:THReshold:LOW.... 2041
BUS<m>:MDIO:COUPling.... 2042
BUS<m>:MDIO:DATA:SOURce.... 2041
BUS<m>:MDIO:DATA:THReshold:HIGH.... 2041
BUS<m>:MDIO:DATA:THReshold:LOW.... 2042
BUS<m>:MDIO:PRESet.... 2042
BUS<m>:MDIO:WCOunt?.... 2045
BUS<m>:MDIO:WORD<n>:DATA?.... 2045
BUS<m>:MDIO:WORD<n>:PHYS?.... 2046
BUS<m>:MDIO:WORD<n>:REGI?.... 2046
BUS<m>:MDIO:WORD<n>:ST?.... 2046
BUS<m>:MDIO:WORD<n>:STARt?.... 2047
BUS<m>:MDIO:WORD<n>:STATe?.... 2047
BUS<m>:MDIO:WORD<n>:STOP?.... 2048
BUS<m>:MDIO:WORD<n>:SYMBol?.... 2048
BUS<m>:MDIO:WORD<n>:TYPE?.... 2048
BUS<m>:MILStd:MAXResponse:BITS.... 1957
BUS<m>:MILStd:MAXResponse:SELect.... 1958
BUS<m>:MILStd:MINGap:BITS.... 1958
BUS<m>:MILStd:MINGap:SELect.... 1958
BUS<m>:MILStd:POLarity.... 1959
BUS<m>:MILStd:PRESet.... 1959
BUS<m>:MILStd:SOURce.... 1957
BUS<m>:MILStd:THReshold:HIGH.... 1959
BUS<m>:MILStd:THReshold:LOW.... 1959
BUS<m>:MILStd:WCOunt?.... 1970
BUS<m>:MILStd:WORD<n>:DATA?.... 1970
BUS<m>:MILStd:WORD<n>:INFO?.... 1971

BUS<m>:MILStd:WORD<n>:RTADdress?.....1971
BUS<m>:MILStd:WORD<n>:STARt?.....1971
BUS<m>:MILStd:WORD<n>:STATus?.....1972
BUS<m>:MILStd:WORD<n>:STOP?.....1972
BUS<m>:MILStd:WORD<n>:SYMBol?.....1972
BUS<m>:MILStd:WORD<n>:TYPE?.....1973
BUS<m>:MPHY:DLANes.....2274
BUS<m>:MPHY:DONE:SOURce.....2275
BUS<m>:MPHY:DONE:THReshold.....2275
BUS<m>:MPHY:DTHRee:SOURce.....2275
BUS<m>:MPHY:DTHRee:THReshold.....2275
BUS<m>:MPHY:DTWO:SOURce.....2275
BUS<m>:MPHY:DTWO:THReshold.....2275
BUS<m>:MPHY:DZERo:SOURce.....2275
BUS<m>:MPHY:DZERo:THReshold.....2275
BUS<m>:MPHY:RESult:FCOunt?.....2294
BUS<m>:MPHY:RESult:FRAMe<n>:CCOunt?.....2295
BUS<m>:MPHY:RESult:FRAMe<n>:CELL<o>:DATA?.....2295
BUS<m>:MPHY:RESult:FRAMe<n>:CELL<o>:STARt?.....2295
BUS<m>:MPHY:RESult:FRAMe<n>:CELL<o>:STATe?.....2296
BUS<m>:MPHY:RESult:FRAMe<n>:CELL<o>:STOP?.....2296
BUS<m>:MPHY:RESult:FRAMe<n>:CELL<o>:TYPE?.....2297
BUS<m>:MPHY:RESult:FRAMe<n>:DATA?.....2297
BUS<m>:MPHY:RESult:FRAMe<n>:FTYPe?.....2293
BUS<m>:MPHY:RESult:FRAMe<n>:INFO?.....2297
BUS<m>:MPHY:RESult:FRAMe<n>:SSFT?.....2298
BUS<m>:MPHY:RESult:FRAMe<n>:STARt?.....2298
BUS<m>:MPHY:RESult:FRAMe<n>:STATe?.....2294
BUS<m>:MPHY:RESult:FRAMe<n>:STOP?.....2298
BUS<m>:MPHY:SCRMode.....2276
BUS<m>:MPHY:SMLanes.....2276
BUS<m>:MPHY:THCoupling.....2276
BUS<m>:MPHY:THPReset.....2276
BUS<m>:MPHY:VARiant.....2274
BUS<m>:NEWList.....1672
BUS<m>:PARallel:BIT<n>:DESKew.....2426
BUS<m>:PARallel:BIT<n>:LABel.....2427
BUS<m>:PARallel:BIT<n>[:STATe].....2424
BUS<m>:PARallel:CLEar.....2429
BUS<m>:PARallel:CLOCk.....2429
BUS<m>:PARallel:CLON.....2429
BUS<m>:PARallel:CLSLope.....2429
BUS<m>:PARallel:DATA:FORMat.....2427
BUS<m>:PARallel:DATA:FORMat.....2441
BUS<m>:PARallel:DATA:HEADer?.....2442
BUS<m>:PARallel:DATA[:VALues]?.....2442
BUS<m>:PARallel:DECTable<n>:COUNt?.....2430
BUS<m>:PARallel:DECTable<n>:DATA?.....2430
BUS<m>:PARallel:DECTable<n>:SHOW.....2430
BUS<m>:PARallel:DESoffset.....2427

BUS<m>:PARallel:DISPlay:BTYP 2428
BUS<m>:PARallel:DISPlay:SHBU 2428
BUS<m>:PARallel:DISPlay:SHDI 2428
BUS<m>:PARallel:HYSTeresis<n> 2426
BUS<m>:PARallel:STATe 2424
BUS<m>:PARallel:TECHnology 2425
BUS<m>:PARallel:THCoupling 2425
BUS<m>:PARallel:THReshold<n> 2424
BUS<m>:PCIE:DSCRambling 2347
BUS<m>:PCIE:FIDLe 2347
BUS<m>:PCIE:GEN 2347
BUS<m>:PCIE:LNKW 2348
BUS<m>:PCIE:LONE:SOURce 2348
BUS<m>:PCIE:LONE:THRHigh 2348
BUS<m>:PCIE:LONE:THRLow 2349
BUS<m>:PCIE:LTHRee:SOURce 2348
BUS<m>:PCIE:LTHRee:THRHigh 2348
BUS<m>:PCIE:LTHRee:THRLow 2349
BUS<m>:PCIE:LTWO:SOURce 2348
BUS<m>:PCIE:LTWO:THRHigh 2348
BUS<m>:PCIE:LTWO:THRLow 2349
BUS<m>:PCIE:LZER:SOURce 2348
BUS<m>:PCIE:LZER:THRHigh 2348
BUS<m>:PCIE:LZER:THRLow 2349
BUS<m>:PCIE:RESult:PACKet<n>:ADDR? 2360
BUS<m>:PCIE:RESult:PACKet<n>:BITRate? 2367
BUS<m>:PCIE:RESult:PACKet<n>:CPID? 2360
BUS<m>:PCIE:RESult:PACKet<n>:CPS? 2360
BUS<m>:PCIE:RESult:PACKet<n>:DATA? 2361
BUS<m>:PCIE:RESult:PACKet<n>:DFC? 2361
BUS<m>:PCIE:RESult:PACKet<n>:FCOunt? 2362
BUS<m>:PCIE:RESult:PACKet<n>:FLD<o>:FVAL? 2362
BUS<m>:PCIE:RESult:PACKet<n>:FLD<o>:STATus? 2362
BUS<m>:PCIE:RESult:PACKet<n>:FLD<o>:TYPE? 2363
BUS<m>:PCIE:RESult:PACKet<n>:FLD<o>:VAL? 2363
BUS<m>:PCIE:RESult:PACKet<n>:HFC? 2364
BUS<m>:PCIE:RESult:PACKet<n>:LEN? 2364
BUS<m>:PCIE:RESult:PACKet<n>:RQID? 2364
BUS<m>:PCIE:RESult:PACKet<n>:SEQN? 2364
BUS<m>:PCIE:RESult:PACKet<n>:STARt? 2365
BUS<m>:PCIE:RESult:PACKet<n>:STATe? 2365
BUS<m>:PCIE:RESult:PACKet<n>:STOP? 2366
BUS<m>:PCIE:RESult:PACKet<n>:TAG? 2366
BUS<m>:PCIE:RESult:PACKet<n>:TYPE? 2366
BUS<m>:PCIE:RESult:PCOunt? 2367
BUS<m>:PCIE:SMLanes 2349
BUS<m>:RESult 1671
BUS<m>:RFFE:CLOCk:SOURce 2055
BUS<m>:RFFE:CLOCk:THReshold:HIGH 2057
BUS<m>:RFFE:CLOCk:THReshold:HYSTeresis 2058

BUS<m>:RFFE:CLOCk:THReshold:LOW....2058
BUS<m>:RFFE:COUPling....2056
BUS<m>:RFFE:DATA:SOURce....2056
BUS<m>:RFFE:DATA:THReshold:HIGH....2056
BUS<m>:RFFE:DATA:THReshold:HYSTeresis....2057
BUS<m>:RFFE:DATA:THReshold:LOW....2057
BUS<m>:RFFE:FCOunt?....2070
BUS<m>:RFFE:GFILter....2058
BUS<m>:RFFE:GFWidth....2059
BUS<m>:RFFE:MINGap:SELect....2059
BUS<m>:RFFE:MINGap:TIME....2059
BUS<m>:RFFE:PRESet....2056
BUS<m>:RFFE:RDMD....2059
BUS<m>:RFFE:SEQuence<n>:ADDRess?....2072
BUS<m>:RFFE:SEQuence<n>:BCOunt?....2072
BUS<m>:RFFE:SEQuence<n>:BYTE<o>:NAME?....2075
BUS<m>:RFFE:SEQuence<n>:BYTE<o>:STATe?....2075
BUS<m>:RFFE:SEQuence<n>:BYTE<o>:VALue?....2075
BUS<m>:RFFE:SEQuence<n>:DATA?....2073
BUS<m>:RFFE:SEQuence<n>:PADone?....2073
BUS<m>:RFFE:SEQuence<n>:PADZero?....2073
BUS<m>:RFFE:SEQuence<n>:PCTRl?....2074
BUS<m>:RFFE:SEQuence<n>:RBTRate?....2074
BUS<m>:RFFE:SEQuence<n>:SADD?....2072
BUS<m>:RFFE:SEQuence<n>:STARt?....2071
BUS<m>:RFFE:SEQuence<n>:STATe?....2070
BUS<m>:RFFE:SEQuence<n>:STOP?....2071
BUS<m>:RFFE:SEQuence<n>:SYMBol?....2073
BUS<m>:RFFE:SEQuence<n>:TYPE?....2071
BUS<m>:RFFE:SEQuence<n>:WBTRate?....2074
BUS<m>:RFFE:VERSion....2060
BUS<m>:SENT:CLKPeriod....1900
BUS<m>:SENT:CLKTolerance....1900
BUS<m>:SENT:CRCMethod....1901
BUS<m>:SENT:CRCVersion....1901
BUS<m>:SENT:DATA:SOURce....1899
BUS<m>:SENT:DATA:THReshold....1899
BUS<m>:SENT:DNIBbles....1901
BUS<m>:SENT:FCOunt?....1909
BUS<m>:SENT:FRAMe<n>:BITRate?....1916
BUS<m>:SENT:FRAMe<n>:CSValue?....1911
BUS<m>:SENT:FRAMe<n>:DATA?....1911
BUS<m>:SENT:FRAMe<n>:IDTYpe?....1911
BUS<m>:SENT:FRAMe<n>:IDValue?....1912
BUS<m>:SENT:FRAMe<n>:NIBBle<o>:STATe?....1912
BUS<m>:SENT:FRAMe<n>:NIBBle<o>:VALue?....1913
BUS<m>:SENT:FRAMe<n>:PAPTicks?....1913
BUS<m>:SENT:FRAMe<n>:SCOM?....1913
BUS<m>:SENT:FRAMe<n>:SDATa?....1914
BUS<m>:SENT:FRAMe<n>:SDEXport?....1914

BUS<m>:SENT:FRAMe<n>:STARt?.... 1910
BUS<m>:SENT:FRAMe<n>:STATus?.... 1910
BUS<m>:SENT:FRAMe<n>:STOP?.... 1910
BUS<m>:SENT:FRAMe<n>:SYMBol?.... 1914
BUS<m>:SENT:FRAMe<n>:SYNCduration?.... 1915
BUS<m>:SENT:FRAMe<n>:TYPE?.... 1915
BUS<m>:SENT:PPFLength.... 1902
BUS<m>:SENT:PPULse.... 1902
BUS<m>:SENT:RDSL.... 1916
BUS<m>:SENT:SFORmat.... 1901
BUS<m>:SENT:TECHnology.... 1900
BUS<m>:SETReflevels.... 1671
BUS<m>:SPI:BORDer.... 1708
BUS<m>:SPI:FCOunt?.... 1716
BUS<m>:SPI:FRAMe<n>:BITRate?.... 1716
BUS<m>:SPI:FRAMe<n>:COUNt?.... 1716
BUS<m>:SPI:FRAMe<n>:DATA?.... 1715
BUS<m>:SPI:FRAMe<n>:STARt?.... 1717
BUS<m>:SPI:FRAMe<n>:STATus?.... 1716
BUS<m>:SPI:FRAMe<n>:STOP?.... 1717
BUS<m>:SPI:FRAMe<n>:WCOunt?.... 1718
BUS<m>:SPI:FRAMe<n>:WORD<o>:FMISo?.... 1719
BUS<m>:SPI:FRAMe<n>:WORD<o>:FMOSi?.... 1720
BUS<m>:SPI:FRAMe<n>:WORD<o>:MISO?.... 1719
BUS<m>:SPI:FRAMe<n>:WORD<o>:MOSI?.... 1720
BUS<m>:SPI:FRAMe<n>:WORD<o>:STARt?.... 1718
BUS<m>:SPI:FRAMe<n>:WORD<o>:STOP?.... 1718
BUS<m>:SPI:FRCondition.... 1712
BUS<m>:SPI:MISO:POLarity.... 1710
BUS<m>:SPI:MISO:SOURce.... 1710
BUS<m>:SPI:MISO:THReshold.... 1711
BUS<m>:SPI:MOSI:POLarity.... 1711
BUS<m>:SPI:MOSI:SOURce.... 1711
BUS<m>:SPI:MOSI:THReshold.... 1711
BUS<m>:SPI:SCLK:SOURce.... 1709
BUS<m>:SPI:SCLK:THReshold.... 1711
BUS<m>:SPI:SSELect:POLarity.... 1710
BUS<m>:SPI:SSELect:SOURce.... 1709
BUS<m>:SPI:SSELect:THReshold.... 1711
BUS<m>:SPI:TIMeout.... 1712
BUS<m>:SPI:WSIZe.... 1709
BUS<m>:SWIRe:BPOSition.... 1999
BUS<m>:SWIRe:COUPling.... 2002
BUS<m>:SWIRe:DATA:HYSTeresis.... 2000
BUS<m>:SWIRe:DATA:SOURce.... 2000
BUS<m>:SWIRe:DATA:THReshold.... 2000
BUS<m>:SWIRe:MGAP.... 2000
BUS<m>:SWIRe:PRESet.... 2002
BUS<m>:SWIRe:RESults:FCOunt.... 2006
BUS<m>:SWIRe:RESults:FRAMe<n>:DATA?.... 2006

BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:FVAL?....2008
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STARt?....2008
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STATus?....2009
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:STOP?....2009
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:TYPE?....2010
BUS<m>:SWIRe:RESults:FRAMe<n>:FLD<o>:VAL?....2010
BUS<m>:SWIRe:RESults:FRAMe<n>:STARt?....2006
BUS<m>:SWIRe:RESults:FRAMe<n>:STATe?....2007
BUS<m>:SWIRe:RESults:FRAMe<n>:STOP?....2007
BUS<m>:SWIRe:RESults:FRAMe<n>:TYPE?....2007
BUS<m>:SWIRe:STRBe:HYSTeresis....2001
BUS<m>:SWIRe:STRBe:SOURce....2001
BUS<m>:SWIRe:STRBe:THReshold....2001
BUS<m>:SWIRe:SYSLect....2002
BUS<m>:SYMBols....1672
BUS<m>:TBTO:ATTN....2116
BUS<m>:TBTO:FDAM....2116
BUS<m>:TBTO:FDAP....2117
BUS<m>:TBTO:FDIF....2117
BUS<m>:TBTO:FTYP....2117
BUS<m>:TBTO:MODE....2118
BUS<m>:TBTO:POLarity....2118
BUS<m>:TBTO:RDAM....2117
BUS<m>:TBTO:RDAP....2118
BUS<m>:TBTO:RDIF....2118
BUS<m>:TBTO:RESult:FCOunt?....2119
BUS<m>:TBTO:RESult:FRAMe<n>:BITRate?....2120
BUS<m>:TBTO:RESult:FRAMe<n>:CRC?....2120
BUS<m>:TBTO:RESult:FRAMe<n>:DATA?....2120
BUS<m>:TBTO:RESult:FRAMe<n>:DESTaddress?....2121
BUS<m>:TBTO:RESult:FRAMe<n>:DSYMbol?....2121
BUS<m>:TBTO:RESult:FRAMe<n>:FTYPe?....2121
BUS<m>:TBTO:RESult:FRAMe<n>:NUMWords?....2122
BUS<m>:TBTO:RESult:FRAMe<n>:SRCaddress?....2122
BUS<m>:TBTO:RESult:FRAMe<n>:SSYMbol?....2122
BUS<m>:TBTO:RESult:FRAMe<n>:STARt?....2123
BUS<m>:TBTO:RESult:FRAMe<n>:STATe?....2123
BUS<m>:TBTO:RESult:FRAMe<n>:STOP?....2124
BUS<m>:TBTO:RESult:FRAMe<n>:TYPE?....2124
BUS<m>:TBTO:RESult:FRAMe<n>:WORD<o>:TYPE?....2124
BUS<m>:TBTO:RESult:FRAMe<n>:WORD<o>:VALue?....2125
BUS<m>:TBTO:RTYP....2118
BUS<m>:TBTO:SYMRate....2119
BUS<m>:THReshold....1671
BUS<m>:TYPE....1670
BUS<m>:TYPE....1675
BUS<m>:UART:BAUDrate....1730
BUS<m>:UART:BITRate....1729
BUS<m>:UART:EWORd....1732
BUS<m>:UART:PACKets....1731

BUS<m>:UART:PARity.....1730
BUS<m>:UART:POLarity.....1730
BUS<m>:UART:RX:SOURce.....1727
BUS<m>:UART:RX:THReshold.....1728
BUS<m>:UART:SBIT.....1731
BUS<m>:UART:SSIZe.....1731
BUS<m>:UART:TECHnology.....1729
BUS<m>:UART:TOUT.....1732
BUS<m>:UART:TX:SOURce.....1728
BUS<m>:UART:TX:THReshold.....1728
BUS<m>:UART:WORD<n>:BITRate?.....1735
BUS<m>:UART:WORD<n>:COUNt?.....1735
BUS<m>:UART:WORD<n>:RXValue?.....1735
BUS<m>:UART:WORD<n>:SOURce?.....1736
BUS<m>:UART:WORD<n>:STARt?.....1736
BUS<m>:UART:WORD<n>:STATe?.....1736
BUS<m>:UART:WORD<n>:TXValue?.....1735
BUS<m>:USB:DATA:SOURce.....2149
BUS<m>:USB:DATA:THReshold.....2150
BUS<m>:USB:DIFFerential:SOURce.....2149
BUS<m>:USB:DIFFerential:THReshold.....2150
BUS<m>:USB:DMINus:SOURce.....2148
BUS<m>:USB:DMINus:THReshold.....2150
BUS<m>:USB:DPLus:SOURce.....2148
BUS<m>:USB:DPLus:THReshold.....2149
BUS<m>:USB:PACKet<n>:ADDRess?.....2164
BUS<m>:USB:PACKet<n>:CRC?.....2164
BUS<m>:USB:PACKet<n>:DATA?.....2165
BUS<m>:USB:PACKet<n>:ENDPoint?.....2165
BUS<m>:USB:PACKet<n>:ET?.....2165
BUS<m>:USB:PACKet<n>:FRAMe?.....2165
BUS<m>:USB:PACKet<n>:PID?.....2163
BUS<m>:USB:PACKet<n>:PORT?.....2166
BUS<m>:USB:PACKet<n>:SC?.....2166
BUS<m>:USB:PACKet<n>:SEU?.....2166
BUS<m>:USB:PACKet<n>:STARt?.....2167
BUS<m>:USB:PACKet<n>:STATus?.....2167
BUS<m>:USB:PACKet<n>:STOP?.....2167
BUS<m>:USB:PCOunt?.....2168
BUS<m>:USB:STRobe:SOURce.....2149
BUS<m>:USB:STRobe:THReshold.....2150
BUS<m>:USB:TECHnology.....2148
BUS<m>:USBPd:DETail.....2219
BUS<m>:USBPd:HYSTeresis.....2219
BUS<m>:USBPd:RESult:FCOunt?.....2227
BUS<m>:USBPd:RESult:FRAMe<n>:DATA?.....2227
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:FVAL?.....2227
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STARt?.....2228
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STATus?.....2228
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:STOP?.....2229

BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:TYPE?....2229
BUS<m>:USBPd:RESult:FRAMe<n>:FLD<o>:VAL?.... 2229
BUS<m>:USBPd:RESult:FRAMe<n>:INFO?.... 2230
BUS<m>:USBPd:RESult:FRAMe<n>:STARt?....2230
BUS<m>:USBPd:RESult:FRAMe<n>:STATe?....2230
BUS<m>:USBPd:RESult:FRAMe<n>:STOP?.... 2231
BUS<m>:USBPd:RESult:FRAMe<n>:TYPE?....2231
BUS<m>:USBPd:SOURce.... 2219
BUS<m>:USBPd:THRBottom....2220
BUS<m>:USBPd:THReshold....2219
BUS<m>:USBPd:THRMid.... 2220
BUS<m>:USBPd:THRTop.... 2220
BUS<m>:USBThree:BITR.... 2191
BUS<m>:USBThree:DFE:COEFficient....2191
BUS<m>:USBThree:DFE:ENBLe....2192
BUS<m>:USBThree:DFE:ESTCoeff....2192
BUS<m>:USBThree:DFE:TCONt.... 2192
BUS<m>:USBThree:POLarity....2188
BUS<m>:USBThree:PROTocol.... 2192
BUS<m>:USBThree:RESult:FCOunt?....2204
BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:FVAL?.... 2208
BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:STARt?....2207
BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:STATus?....2206
BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:STOP?.... 2207
BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:TYPE?....2206
BUS<m>:USBThree:RESult:FRAMe<n>:FLD<o>:VAL?.... 2207
BUS<m>:USBThree:RESult:FRAMe<n>:INFO?.... 2204
BUS<m>:USBThree:RESult:FRAMe<n>:STARt?....2205
BUS<m>:USBThree:RESult:FRAMe<n>:STATe?....2205
BUS<m>:USBThree:RESult:FRAMe<n>:STOP?.... 2206
BUS<m>:USBThree:RESult:FRAMe<n>:TYPE?....2204
BUS<m>:USBThree:SCRMode....2189
BUS<m>:USBThree:SOURce.... 2188
BUS<m>:USBThree:THRHigh....2189
BUS<m>:USBThree:THRLow....2189
BUS<m>:USBThree:TYPE.... 2189
BUS<m>:ZCOupling.... 1672
BUS<m>[:STATe].... 1670
CALCulate:MATH<m>:ARIThmetics....1457
CALCulate:MATH<m>:DATA:HEADer?.... 1460
CALCulate:MATH<m>:DATA:HEADer?.... 1579
CALCulate:MATH<m>:DATA:STYPe?.... 1459
CALCulate:MATH<m>:DATA:STYPe?.... 1578
CALCulate:MATH<m>:DATA[:VALues]?.... 1460
CALCulate:MATH<m>:DATA[:VALues]?.... 1579
CALCulate:MATH<m>:ENVSelection.... 1457
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:ADJusted?.... 1568
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:AUTO....1569
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution]:RATio.... 1569
CALCulate:MATH<m>:FFT:BANDwidth[:RESolution][:VALue].... 1569

CALCulate:MATH<m>:FFT:CFRequency.....1567
CALCulate:MATH<m>:FFT:COUPled:WITH<1..8>.....1578
CALCulate:MATH<m>:FFT:FRAMe:ARIThmetics.....1571
CALCulate:MATH<m>:FFT:FRAMe:COVerage?.....1571
CALCulate:MATH<m>:FFT:FRAMe:MAXCount.....1571
CALCulate:MATH<m>:FFT:FRAMe:OFACtor.....1572
CALCulate:MATH<m>:FFT:FULLspan.....1567
CALCulate:MATH<m>:FFT:GATE:ABSolute:STARt.....1573
CALCulate:MATH<m>:FFT:GATE:ABSolute:STOP.....1573
CALCulate:MATH<m>:FFT:GATE:COUPling.....1572
CALCulate:MATH<m>:FFT:GATE:MODE.....1573
CALCulate:MATH<m>:FFT:GATE:RELative:STARt.....1574
CALCulate:MATH<m>:FFT:GATE:RELative:STOP.....1574
CALCulate:MATH<m>:FFT:GATE:ZCOupling.....1574
CALCulate:MATH<m>:FFT:GATE[:STATe].....1575
CALCulate:MATH<m>:FFT:LOGScale.....1566
CALCulate:MATH<m>:FFT:MAGNitude:LEVel.....1575
CALCulate:MATH<m>:FFT:MAGNitude:RANGe.....1575
CALCulate:MATH<m>:FFT:MAGNitude:SCALe.....1575
CALCulate:MATH<m>:FFT:PHASe:SCALe.....1577
CALCulate:MATH<m>:FFT:PHASe:SUPPression.....1577
CALCulate:MATH<m>:FFT:PHASe:THReshold.....1577
CALCulate:MATH<m>:FFT:PHASe:UNWRap.....1577
CALCulate:MATH<m>:FFT:SPAN.....1568
CALCulate:MATH<m>:FFT:SPECtrogram:CMODe.....1580
CALCulate:MATH<m>:FFT:SPECtrogram:STATe.....1581
CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:POSition.....1581
CALCulate:MATH<m>:FFT:SPECtrogram:TIMeline<n>:STATe.....1581
CALCulate:MATH<m>:FFT:STARt.....1566
CALCulate:MATH<m>:FFT:STOP.....1567
CALCulate:MATH<m>:FFT:TYPE.....1568
CALCulate:MATH<m>:FFT:USEColtab.....1580
CALCulate:MATH<m>:FFT:WINDow:TYPE.....1569
CALCulate:MATH<m>:STATe.....1456
CALCulate:MATH<m>:VERTical:MAXimum.....1459
CALCulate:MATH<m>:VERTical:OFFSet.....1458
CALCulate:MATH<m>:VERTical:RANGe.....1458
CALCulate:MATH<m>:VERTical:SCALe.....1459
CALCulate:MATH<m>[:EXPRession][:DEFine].....1456
CALibration:DATE?.....1322
CALibration:DATE?.....2571
CALibration:RESult?.....1323
CALibration:RESult?.....2571
CALibration:SOURce:FREQuency.....1386
CALibration:SOURce:STATe.....1386
CALibration:TIME?.....1322
CALibration:TIME?.....2571
CDR:HARDware:BITRate.....2516
CDR:HARDware:BREStimate.....2516
CDR:HARDware:ESLope.....2516

CDR:HARDware:PLL:BWIDth.... 2517
CDR:HARDware:PLL:DAMPing.... 2517
CDR:HARDware:PLL:ORDer.... 2516
CDR:HARDware:PLL:RELBwidth.... 2517
CDR:HARDware:SAMTime.... 2517
CDR:HARDware:SAMTime.... 2519
CDR:HARDware:SERStandard.... 2516
CDR:HARDware:SOURce.... 2515
CDR:SOFTware<m>:ALGorithm.... 2510
CDR:SOFTware<m>:BITRate.... 2512
CDR:SOFTware<m>:ESBRate:BREStimate.... 2511
CDR:SOFTware<m>:ESBRate:CONTinuous.... 2511
CDR:SOFTware<m>:ESBRate:SOURce.... 2511
CDR:SOFTware<m>:ESLope.... 2512
CDR:SOFTware<m>:PLL:BWIDth.... 2514
CDR:SOFTware<m>:PLL:DAMPing.... 2514
CDR:SOFTware<m>:PLL:ORDer.... 2513
CDR:SOFTware<m>:PLL:RELBwidth.... 2514
CDR:SOFTware<m>:RESults.... 2513
CDR:SOFTware<m>:REVLockpll.... 2514
CDR:SOFTware<m>:SAMTime.... 2515
CDR:SOFTware<m>:SERStandard.... 2510
CDR:SOFTware<m>:SYNC.... 2512
CHANnel<m>:BANDwidth.... 1340
CHANnel<m>:COUPling.... 1337
CHANnel<m>:CPLing.... 1340
CHANnel<m>:DIGFilter:CUToff.... 1383
CHANnel<m>:DIGFilter:STATe.... 1383
CHANnel<m>:EATScale.... 1346
CHANnel<m>:EATTenuation.... 1346
CHANnel<m>:EXPortstate.... 1651
CHANnel<m>:FILDesign.... 1341
CHANnel<m>:GND.... 1338
CHANnel<m>:IMPedance.... 1340
CHANnel<m>:INVert.... 1339
CHANnel<m>:IQ:CFRequency.... 2475
CHANnel<m>:IQ:DATA:HEADer?.... 2477
CHANnel<m>:IQ:DATA[:VALues]?.... 2477
CHANnel<m>:IQ:INPMode.... 2473
CHANnel<m>:IQ:INPType.... 2473
CHANnel<m>:IQ:SBIF.... 2476
CHANnel<m>:IQ:SBRF.... 2475
CHANnel<m>:IQ:WARNing:MESSage?.... 2476
CHANnel<m>:IQ:WARNing[:STATe]?.... 2476
CHANnel<m>:OFFSet.... 1339
CHANnel<m>:OVERload.... 1341
CHANnel<m>:POSition.... 1339
CHANnel<m>:RANGe.... 1338
CHANnel<m>:SCALe.... 1338
CHANnel<m>:SKEW:EXTended.... 1385

CHANnel<m>:SKEW:MANual.... 1384
CHANnel<m>:SKEW:TIME.... 1384
CHANnel<m>:STATe.... 1337
CHANnel<m>[:WAVeform<n>]:ARIThmetics.... 1334
CHANnel<m>[:WAVeform<n>]:DATA:HEADer?.... 1342
CHANnel<m>[:WAVeform<n>]:DATA[:VALues]?.... 1342
CHANnel<m>[:WAVeform<n>]:HISTory:CURRent.... 1461
CHANnel<m>[:WAVeform<n>]:HISTory:PLAY.... 1463
CHANnel<m>[:WAVeform<n>]:HISTory:REPLay.... 1463
CHANnel<m>[:WAVeform<n>]:HISTory:STARt.... 1462
CHANnel<m>[:WAVeform<n>]:HISTory:STOP.... 1462
CHANnel<m>[:WAVeform<n>]:HISTory:TPACq.... 1463
CHANnel<m>[:WAVeform<n>]:HISTory:TSABsolute?.... 1464
CHANnel<m>[:WAVeform<n>]:HISTory:TSDate?.... 1464
CHANnel<m>[:WAVeform<n>]:HISTory:TSRelative?.... 1464
CHANnel<m>[:WAVeform<n>]:HISTory:TSRReference?.... 1465
CHANnel<m>[:WAVeform<n>]:HISTory[:STATe].... 1461
CHANnel<m>[:WAVeform<n>]:TYPE.... 1333
CHANnel<m>[:WAVeform<n>][:STATe].... 1333
CURSor<m>:AOFF.... 1469
CURSor<m>:FFT:SETCenter.... 1478
CURSor<m>:FFT:TOCenter.... 1478
CURSor<m>:FUNCtion.... 1469
CURSor<m>:LABel:X<1..2>.... 1475
CURSor<m>:LABel:Y<1..2>.... 1476
CURSor<m>:LABel[:LABel].... 1475
CURSor<m>:MAXimum:LEFT.... 1478
CURSor<m>:MAXimum:NEXT.... 1479
CURSor<m>:MAXimum:RIGHt.... 1478
CURSor<m>:MAXimum[:PEAK].... 1478
CURSor<m>:PEXCursion.... 1630
CURSor<m>:SIAD.... 1476
CURSor<m>:SOURce.... 1470
CURSor<m>:SSOurce.... 1470
CURSor<m>:STATe.... 1469
CURSor<m>:STYLe.... 1314
CURSor<m>:THReshold.... 1631
CURSor<m>:TRACking[:STATe].... 1472
CURSor<m>:USSource.... 1471
CURSor<m>:X1ENvelope.... 1474
CURSor<m>:X1Position.... 1472
CURSor<m>:X2ENvelope.... 1475
CURSor<m>:X2Position.... 1472
CURSor<m>:XCOupling.... 1473
CURSor<m>:XDELta:INVerse?.... 1477
CURSor<m>:XDELta[:VALue]?.... 1476
CURSor<m>:Y1Position.... 1473
CURSor<m>:Y2Position.... 1473
CURSor<m>:YCOupling.... 1474
CURSor<m>:YDELta:SLOPe.... 1477

CURSor<m>:YDELta[:VALue]?.... 1477
DEEMbedding<m>:ADD.... 2559
DEEMbedding<m>:BANDwidth.... 2560
DEEMbedding<m>:CADC?.... 2568
DEEMbedding<m>:CCOunt?.... 2560
DEEMbedding<m>:CMPDelay.... 2561
DEEMbedding<m>:COMPonent<n>:IDENtical.... 2567
DEEMbedding<m>:COMPonent<n>:IPONe.... 2567
DEEMbedding<m>:COMPonent<n>:IPTWo.... 2567
DEEMbedding<m>:COMPonent<n>:LOAD:IMPedance?.... 2565
DEEMbedding<m>:COMPonent<n>:LOAD:REMProbeload.... 2565
DEEMbedding<m>:COMPonent<n>:MODE.... 2564
DEEMbedding<m>:COMPonent<n>:NAME.... 2563
DEEMbedding<m>:COMPonent<n>:OPONe.... 2567
DEEMbedding<m>:COMPonent<n>:OPTWo.... 2567
DEEMbedding<m>:COMPonent<n>:PCOunt?.... 2564
DEEMbedding<m>:COMPonent<n>:SPONe.... 2568
DEEMbedding<m>:COMPonent<n>:SPTWo.... 2568
DEEMbedding<m>:COMPonent<n>:TYPE.... 2563
DEEMbedding<m>:COMPonent<n>[:STATe].... 2562
DEEMbedding<m>:DELete.... 2562
DEEMbedding<m>:MATTenuation?.... 2569
DEEMbedding<m>:MGAin?.... 2569
DEEMbedding<m>:NAME.... 2561
DEEMbedding<m>:OPEN.... 2562
DEEMbedding<m>:PRESet.... 2561
DEEMbedding<m>:REMove.... 2560
DEEMbedding<m>:SAVE.... 2562
DEEMbedding<m>[:STATe].... 2559
DIAGnostic:SERVice:CHANnelcount?.... 1324
DIAGnostic:SERVice:COMPutername.... 1297
DIAGnostic:SERVice:FWVersion?.... 1324
DIAGnostic:SERVice:PARTnumber.... 1324
DIAGnostic:SERVice:PWD.... 2572
DIAGnostic:SERVice:SERialnumber?.... 1324
DIAGnostic:SERVice:STST:EXECute.... 2571
DIAGnostic:SERVice:STST:STATe?.... 1323
DIAGnostic:SERVice:STST:STATe?.... 2571
DIAGnostic:SERVice:WFAModel?.... 2570
DIAGnostic:SERVice:WFASeries?.... 2570
DIAGnostic:SERVice:WFAType?.... 2570
DIFFerential<m>:AOUTput.... 1380
DIFFerential<m>:BOUTput.... 1380
DIFFerential<m>:COMMon:OFFSet.... 1382
DIFFerential<m>:COMMon:POSition.... 1382
DIFFerential<m>:COMMon:SCALe.... 1381
DIFFerential<m>:COMMon:STATe.... 1381
DIFFerential<m>:COUPling.... 1381
DIFFerential<m>:DIFFerential:OFFSet.... 1382
DIFFerential<m>:DIFFerential:POSition.... 1382

DIFFerential<m>:DIFFerential:SCALe....1381
DIFFerential<m>:DIFFerential:STATe....1381
DIFFerential<m>:NSIGnal:STATe....1381
DIFFerential<m>:NSIGnal[:SELect]....1380
DIFFerential<m>:PSIGnal:STATe....1381
DIFFerential<m>:PSIGnal[:SELect]....1380
DIFFerential<m>:STATe....1380
DIGital<m>:DATA:HEADer?....2441
DIGital<m>:DATA[:VALues]?....2441
DIGital<m>:DESKew....2423
DIGital<m>:DISPlay....2421
DIGital<m>:HYSTeresis....2422
DIGital<m>:LABel....2423
DIGital<m>:TECHnology....2421
DIGital<m>:THCoupling....2422
DIGital<m>:THReshold....2421
DISPlay:CLR....1321
DISPlay:COLor:PALette:ADD....1298
DISPlay:COLor:PALette:COUNt?....1299
DISPlay:COLor:PALette:POINt:ADD....1299
DISPlay:COLor:PALette:POINt:COUNt?....1300
DISPlay:COLor:PALette:POINt:INSert....1299
DISPlay:COLor:PALette:POINt:REMove....1299
DISPlay:COLor:PALette:POINt[:VALue]....1299
DISPlay:COLor:PALette:REMove....1298
DISPlay:COLor:SIGNal<m>:ASSign....1297
DISPlay:COLor:SIGNal<m>:COLor....1297
DISPlay:COLor:SIGNal<m>:USE....1298
DISPlay:DIAGram:CROSshair....1301
DISPlay:DIAGram:FINegrid....1301
DISPlay:DIAGram:GRID....1300
DISPlay:DIAGram:LABels....1300
DISPlay:DIAGram:STYLe....1321
DISPlay:DIAGram:TITLe....1306
DISPlay:DIAGram:XFIXed....1301
DISPlay:DIAGram:YFIXed....1301
DISPlay:EXTended:PORDialogs....1313
DISPlay:EXTended:POSDialogs....1313
DISPlay:GATE:TRANsparency....1306
DISPlay:INTensity....1320
DISPlay:PERSistence:INFinite....1320
DISPlay:PERSistence:RESet....1320
DISPlay:PERSistence:TIME....1320
DISPlay:PERSistence[:STATe]....1319
DISPlay:RESultboxes:CUPosition....1314
DISPlay:RESultboxes:DEPosition....1673
DISPlay:RESultboxes:MEPosition....1315
DISPlay:SIGBar[:STATe]....1306
DISPlay:SIGNal:LABel:ADD....1308
DISPlay:SIGNal:LABel:FONTsize....1308

DISPlay:SIGNal:LABel:HORizontal:ABSolute:POSition.....1312
DISPlay:SIGNal:LABel:HORizontal:RELative:POSition.....1313
DISPlay:SIGNal:LABel:IDSList.....1310
DISPlay:SIGNal:LABel:POSMode.....1307
DISPlay:SIGNal:LABel:REMove.....1309
DISPlay:SIGNal:LABel:SHOW.....1310
DISPlay:SIGNal:LABel:STATe.....1311
DISPlay:SIGNal:LABel:TEXT.....1312
DISPlay:SIGNal:LABel:VERTical:ABSolute:POSition.....1312
DISPlay:SIGNal:LABel:VERTical:RELative:POSition.....1313
DISPlay:TRIGger:LINes.....1392
EXECutable:NAME.....1668
EXECutable:PARameter.....1669
EXECutable:WDIRectory.....1669
EXPort:EYE:DATA?.....1658
EXPort:EYE:NAME.....1658
EXPort:EYE:SAVE.....1658
EXPort:EYE:SELect.....1658
EXPort:EYE:SOURce?.....1658
EXPort:HISTogram:DATA?.....1657
EXPort:HISTogram:INCidence.....1656
EXPort:HISTogram:NAME.....1657
EXPort:HISTogram:SAVE.....1657
EXPort:HISTogram:SELect.....1656
EXPort:MEASurement:DATA?.....1660
EXPort:MEASurement:NAME.....1659
EXPort:MEASurement:SAVE.....1660
EXPort:MEASurement:SELect.....1659
EXPort:MEASurement:TYPE.....1659
EXPort:RESult:DECSymbol.....1668
EXPort:RESult:DELimiter.....1668
EXPort:WAVeform:CURSorset.....1653
EXPort:WAVeform:DISPlayoff.....1656
EXPort:WAVeform:DLOGging.....1654
EXPort:WAVeform:FASTexport.....1656
EXPort:WAVeform:INCXvalues.....1655
EXPort:WAVeform:MEAS.....1653
EXPort:WAVeform:MULTichannel.....1650
EXPort:WAVeform:NAME.....1651
EXPort:WAVeform:RAW.....1655
EXPort:WAVeform:SAVE.....1651
EXPort:WAVeform:SCOPe.....1652
EXPort:WAVeform:SOURce.....1649
EXPort:WAVeform:STARt.....1652
EXPort:WAVeform:STOP.....1652
EXPort:WAVeform:TIMestamps.....1654
EXPort:WAVeform:ZOOM.....1653
EYE<m>:DISPlay:MODE.....2408
EYE<m>:DISPlay:PERSistence:INFinite.....2409
EYE<m>:DISPlay:PERSistence:RESet.....2409

EYE<m>:DISPlay:PERSistence:STATe....2409
EYE<m>:DISPlay:PERSistence:TIME....2409
EYE<m>:FILTer:BPATtern:CENTer....2415
EYE<m>:FILTer:BPATtern:MODE....2414
EYE<m>:FILTer:BPATtern:PREFix<1..8>....2415
EYE<m>:FILTer:BPATtern:RESet....2415
EYE<m>:FILTer:BPATtern:SLEFt....2415
EYE<m>:FILTer:BPATtern:SRIGht....2415
EYE<m>:FILTer:BPATtern:SUFFix<1..5>....2415
EYE<m>:FILTer:DDR:FTYPe....2415
EYE<m>:FILTer:DDR:LENerror....2416
EYE<m>:FILTer:STATe....2414
EYE<m>:HORizontal:ABSolute:POSition....2404
EYE<m>:HORizontal:ABSolute:RANGe....2403
EYE<m>:HORizontal:UINTerval:POSition....2404
EYE<m>:HORizontal:UINTerval:RANGe....2404
EYE<m>:HORizontal[:UNIT]....2403
EYE<m>:MEASurement:MAXPrecision....2408
EYE<m>:MSLices....2408
EYE<m>:QUALify:GATE:ABSolute:STARt....2411
EYE<m>:QUALify:GATE:ABSolute:STOP....2411
EYE<m>:QUALify:GATE:COUPling....2410
EYE<m>:QUALify:GATE:CURSor....2412
EYE<m>:QUALify:GATE:MODE....2410
EYE<m>:QUALify:GATE:RELative:STARt....2411
EYE<m>:QUALify:GATE:RELative:STOP....2411
EYE<m>:QUALify:GATE:ZDIagram....2412
EYE<m>:QUALify:GATE[:STATe]....2410
EYE<m>:QUALify:SIGNal:CONDition....2413
EYE<m>:QUALify:SIGNal:LSELect....2413
EYE<m>:QUALify:SIGNal:SOURce....2413
EYE<m>:QUALify:SIGNal[:STATe]....2412
EYE<m>:RESult:SLICe<n>:COUNt?....2416
EYE<m>:RESult:SLICe<n>:LPOSition?....2416
EYE<m>:RESult:SLICe<n>[:POSition]?....2417
EYE<m>:RESult:TACQs?....2417
EYE<m>:RESult:TSLices?....2417
EYE<m>:SOURce....2403
EYE<m>:TIMReference:CLOCk:EDGE....2405
EYE<m>:TIMReference:CLOCk:FACTor....2405
EYE<m>:TIMReference:CLOCk:FREQuency....2405
EYE<m>:TIMReference:CLOCk:FREStimation....2406
EYE<m>:TIMReference:CLOCk:LEVel....2406
EYE<m>:TIMReference:CLOCk:OFEStimation....2406
EYE<m>:TIMReference:CLOCk:OFFSet....2406
EYE<m>:TIMReference:CLOCk:SAMTime....2407
EYE<m>:TIMReference:CLOCk:SOURce....2407
EYE<m>:TIMReference:PROTocol:SELect....2407
EYE<m>:TIMReference[:SOURce]....2407
EYE<m>:ZCOupling....2419

EYE<m>:ZDIagram....2420
EYE<m>[:STATe]....2403
FORMat:BORDer....1293
FORMat:BORDer....1387
FORMat:BPATtern....1294
FORMat[:DATA]....1292
GENerator:ALIGnment:DC:ABORt....2461
GENerator:ALIGnment:DC:RESult:DATE?....2461
GENerator:ALIGnment:DC:RESult:TIME?....2461
GENerator:ALIGnment:DC:RESult[:STATe]?....2461
GENerator:ALIGnment:DC[:STARt]....2461
GENerator:SYNC[:COMBination]....2463
GPIB:ADDRess....1294
GPIB:TERMinator....1295
HCOPy:CMAP<m>:DEFault....1663
HCOPy:DESTination<1..2>....1661
HCOPy:DEVice<m>:INVerse....1662
HCOPy:DEVice<m>:LANGuage....1662
HCOPy:IMMediate<m>:NEXT....1664
HCOPy:IMMediate<m>[:DUM]....1664
HCOPy:ISBA....1663
HCOPy:SSD....1663
HCOPy:WBKG....1662
HDEFinition:BWIDth....1387
HDEFinition:RESolution?....1387
HDEFinition:STATe....1387
IQ:BWIDth?....2474
IQ:INSRate....2473
IQ:RBWidth....2474
IQ:RLENgth....2474
IQ:SRATe....2474
IQ:STATe....2473
LAYout:ADD....1302
LAYout:HISTogram:ADD....1521
LAYout:HISTogram:ENABle....1522
LAYout:HISTogram:HORZ:ABSolute:STARt....1524
LAYout:HISTogram:HORZ:ABSolute:STOP....1524
LAYout:HISTogram:HORZ:MODE....1524
LAYout:HISTogram:HORZ:RELative:STARt....1525
LAYout:HISTogram:HORZ:RELative:STOP....1525
LAYout:HISTogram:MODE....1523
LAYout:HISTogram:REMove....1527
LAYout:HISTogram:RESet....1527
LAYout:HISTogram:SOURce....1523
LAYout:HISTogram:VERTical:ABSolute:STARt....1526
LAYout:HISTogram:VERTical:ABSolute:STOP....1526
LAYout:HISTogram:VERTical:MODE....1525
LAYout:HISTogram:VERTical:RELative:STARt....1526
LAYout:HISTogram:VERTical:RELative:STOP....1526
LAYout:REMove....1304

LAYout:SHOW....1304
LAYout:SIGNal:ASSign....1304
LAYout:SIGNal:AXIS....1466
LAYout:SIGNal:UNASsign....1306
LAYout:ZOOM:ADD....1438
LAYout:ZOOM:ADDCoupled....1438
LAYout:ZOOM:HORZ:ABSolute:POSition....1439
LAYout:ZOOM:HORZ:ABSolute:SPAN....1440
LAYout:ZOOM:HORZ:ABSolute:STARt....1440
LAYout:ZOOM:HORZ:ABSolute:STOP....1441
LAYout:ZOOM:HORZ:MODE....1439
LAYout:ZOOM:HORZ:RELative:POSition....1441
LAYout:ZOOM:HORZ:RELative:SPAN....1441
LAYout:ZOOM:HORZ:RELative:STARt....1442
LAYout:ZOOM:HORZ:RELative:STOP....1442
LAYout:ZOOM:ONEDiagram....1307
LAYout:ZOOM:POSCoupling....1439
LAYout:ZOOM:REMove....1446
LAYout:ZOOM:VERTical:ABSolute:POSition....1443
LAYout:ZOOM:VERTical:ABSolute:SPAN....1443
LAYout:ZOOM:VERTical:ABSolute:STARt....1444
LAYout:ZOOM:VERTical:ABSolute:STOP....1444
LAYout:ZOOM:VERTical:MODE....1442
LAYout:ZOOM:VERTical:RELative:POSition....1444
LAYout:ZOOM:VERTical:RELative:SPAN....1445
LAYout:ZOOM:VERTical:RELative:STARt....1445
LAYout:ZOOM:VERTical:RELative:STOP....1445
MEASurement<m>:ADDitional....1484
MEASurement<m>:AMPTime:ALEVel....1493
MEASurement<m>:AMPTime:ARIThmetic:CONStant....1500
MEASurement<m>:AMPTime:ARIThmetic:COUNit....1500
MEASurement<m>:AMPTime:ARIThmetic:CUNState....1501
MEASurement<m>:AMPTime:ARIThmetic:CUUNit....1501
MEASurement<m>:AMPTime:ARIThmetic:CUUState....1502
MEASurement<m>:AMPTime:ARIThmetic:OPERator....1502
MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:AMPTime....1502
MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:GROup....1503
MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:SPECtrum....1503
MEASurement<m>:AMPTime:ARIThmetic:PARameter<n>:TYPE....1504
MEASurement<m>:AMPTime:CLCK<n>:LSELect....1497
MEASurement<m>:AMPTime:CSLope....1496
MEASurement<m>:AMPTime:DATA<n>:LSELect....1497
MEASurement<m>:AMPTime:DELay<n>:DIRection....1494
MEASurement<m>:AMPTime:DELay<n>:ECOunt....1495
MEASurement<m>:AMPTime:DELay<n>:LSELect....1495
MEASurement<m>:AMPTime:DELay<n>:SLOPe....1495
MEASurement<m>:AMPTime:DTOTrigger<n>:LSELect....1498
MEASurement<m>:AMPTime:DTOTrigger<n>:SLOPe....1497
MEASurement<m>:AMPTime:ESLope....1496
MEASurement<m>:AMPTime:LCHeck<n>:LOWer:LIMit....1499

MEASurement<m>:AMPTime:LCHeck<n>:LOWer:MARGin....1499
MEASurement<m>:AMPTime:LCHeck<n>:UPPer:LIMit....1499
MEASurement<m>:AMPTime:LCHeck<n>:UPPer:MARGin....1499
MEASurement<m>:AMPTime:LCHeck<n>:VALid....1498
MEASurement<m>:AMPTime:PFSLope....1493
MEASurement<m>:AMPTime:PSLope....1494
MEASurement<m>:AMPTime:PTCount....1496
MEASurement<m>:AOFF....1485
MEASurement<m>:AON....1485
MEASurement<m>:ARES?....1486
MEASurement<m>:ARNames....1486
MEASurement<m>:CATegory....1482
MEASurement<m>:CLEar....1533
MEASurement<m>:DETThreshold....1493
MEASurement<m>:DISPlay:GROuping....1315
MEASurement<m>:DISPlay:HISTogram....1532
MEASurement<m>:DISPlay:LEVels....1531
MEASurement<m>:DISPlay:RESults....1532
MEASurement<m>:DISPlay:STYLe....1532
MEASurement<m>:ENVSelect....1492
MEASurement<m>:EYEJitter:AUToscale....1509
MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:LIMit....1510
MEASurement<m>:EYEJitter:LCHeck<n>:LOWer:MARGin....1511
MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:LIMit....1510
MEASurement<m>:EYEJitter:LCHeck<n>:UPPer:MARGin....1511
MEASurement<m>:EYEJitter:LCHeck<n>:VALid....1510
MEASurement<m>:FSRC....1482
MEASurement<m>:GATE:ABSolute:STARt....1547
MEASurement<m>:GATE:ABSolute:STOP....1547
MEASurement<m>:GATE:CCOupling....1548
MEASurement<m>:GATE:CURSor....1548
MEASurement<m>:GATE:GCOupling....1549
MEASurement<m>:GATE:MODE....1547
MEASurement<m>:GATE:RELative:STARt....1547
MEASurement<m>:GATE:RELative:STOP....1547
MEASurement<m>:GATE:ZCOupling....1548
MEASurement<m>:GATE:ZDIagram....1549
MEASurement<m>:GATE[:STATe]....1546
MEASurement<m>:HISTogram:LCHeck<n>:LOWer:LIMit....1530
MEASurement<m>:HISTogram:LCHeck<n>:LOWer:MARGin....1531
MEASurement<m>:HISTogram:LCHeck<n>:UPPer:LIMit....1530
MEASurement<m>:HISTogram:LCHeck<n>:UPPer:MARGin....1531
MEASurement<m>:HISTogram:LCHeck<n>:VALid....1530
MEASurement<m>:HISTogram:PROBability:LIMit....1529
MEASurement<m>:HISTogram:PROBability:TYPE....1529
MEASurement<m>:HISTogram:SELect....1528
MEASurement<m>:JITTer:CCSLope....2480
MEASurement<m>:JITTer:CDRMode....2482
MEASurement<m>:JITTer:DRUNit....2483
MEASurement<m>:JITTer:NCYCles....2481

MEASurement<m>:JITTer:PULSe....2481
MEASurement<m>:JITTer:SKWRelation....2484
MEASurement<m>:JITTer:SKWSlope....2483
MEASurement<m>:JITTer:SOURce<n>:TIESlope....2482
MEASurement<m>:LCHeck....1549
MEASurement<m>:LTMeas:COUNt....1538
MEASurement<m>:LTMeas:TIME....1539
MEASurement<m>:LTMeas[:STATe]....1538
MEASurement<m>:MAIN....1483
MEASurement<m>:MNOMeas....1534
MEASurement<m>:MULTiple....1534
MEASurement<m>:ONViolation:ACQStop....1550
MEASurement<m>:ONViolation:BEEP....1550
MEASurement<m>:ONViolation:REPort....1551
MEASurement<m>:ONViolation:RUNexec....1551
MEASurement<m>:ONViolation:TRIGgerout....1551
MEASurement<m>:ONViolation:WFMSave....1551
MEASurement<m>:OPTical:LCHeck<n>:LOWer:LIMit....1520
MEASurement<m>:OPTical:LCHeck<n>:LOWer:MARGin....1520
MEASurement<m>:OPTical:LCHeck<n>:UPPer:LIMit....1520
MEASurement<m>:OPTical:LCHeck<n>:UPPer:MARGin....1520
MEASurement<m>:OPTical:LCHeck<n>:VALid....1520
MEASurement<m>:OPTical:OMAMplitude:RELative:STARt....1519
MEASurement<m>:OPTical:OMAMplitude:RELative:STOP....1519
MEASurement<m>:OPTical:UNIT....1519
MEASurement<m>:PROTocol:F2FRame:FLDFrom....1540
MEASurement<m>:PROTocol:F2FRame:FLDTo....1540
MEASurement<m>:PROTocol:F2FRame:FRMFrom....1541
MEASurement<m>:PROTocol:F2FRame:FRMTo....1541
MEASurement<m>:PROTocol:F2FRame:VALFrom....1541
MEASurement<m>:PROTocol:F2FRame:VALTo....1541
MEASurement<m>:PROTocol:FLDValue:FLD....1542
MEASurement<m>:PROTocol:FLDValue:FRM....1542
MEASurement<m>:PROTocol:FLDValue:TRCK....1542
MEASurement<m>:PROTocol:FLDValue:VAL....1542
MEASurement<m>:PROTocol:MBITrate:FLD....1543
MEASurement<m>:PROTocol:MBITrate:FRM....1543
MEASurement<m>:PROTocol:MBITrate:VAL....1543
MEASurement<m>:PROTocol:SBITrate:FLD....1543
MEASurement<m>:PROTocol:SBITrate:FRM....1543
MEASurement<m>:PROTocol:SBITrate:VAL....1544
MEASurement<m>:PROTocol:T2FRame:DIRection....1544
MEASurement<m>:PROTocol:T2FRame:FLD....1544
MEASurement<m>:PROTocol:T2FRame:FRM....1544
MEASurement<m>:PROTocol:T2FRame:VALue....1545
MEASurement<m>:REFLevel:RESult:BINNer?....1564
MEASurement<m>:REFLevel:RESult:BOUTer?....1564
MEASurement<m>:REFLevel:RESult:LOWer?....1563
MEASurement<m>:REFLevel:RESult:MIDDle?....1563
MEASurement<m>:REFLevel:RESult:SIGHigh?....1563

MEASurement<m>:REFLevel:RESult:SIGLow?....1563
MEASurement<m>:REFLevel:RESult:TINNer?....1564
MEASurement<m>:REFLevel:RESult:TOUTer?....1565
MEASurement<m>:REFLevel:RESult:UPPer?....1563
MEASurement<m>:RESult:AVG?....1487
MEASurement<m>:RESult:COUNt?....1489
MEASurement<m>:RESult:EVTCount?....1487
MEASurement<m>:RESult:INVerse....1316
MEASurement<m>:RESult:LABorder....1315
MEASurement<m>:RESult:MAXCount....1515
MEASurement<m>:RESult:NPEak?....1487
MEASurement<m>:RESult:PPEak?....1487
MEASurement<m>:RESult:RELiability?....1487
MEASurement<m>:RESult:RMS?....1487
MEASurement<m>:RESult:SHFRequency....1516
MEASurement<m>:RESult:SHLabels....1516
MEASurement<m>:RESult:STARt?....1489
MEASurement<m>:RESult:STDDev?....1487
MEASurement<m>:RESult:STOP?....1489
MEASurement<m>:RESult:WFMCount?....1487
MEASurement<m>:RESult[:ACTual]?....1487
MEASurement<m>:SOURce....1480
MEASurement<m>:SPECtrum:ARIThmetic:CONStant....1504
MEASurement<m>:SPECtrum:ARIThmetic:COUNit....1505
MEASurement<m>:SPECtrum:ARIThmetic:CUNState....1505
MEASurement<m>:SPECtrum:ARIThmetic:CUUNit....1505
MEASurement<m>:SPECtrum:ARIThmetic:CUUState....1506
MEASurement<m>:SPECtrum:ARIThmetic:OPERator....1506
MEASurement<m>:SPECtrum:ARIThmetic:PARameter<n>:AMPTime....1506
MEASurement<m>:SPECtrum:ARIThmetic:PARameter<n>:GROup....1507
MEASurement<m>:SPECtrum:ARIThmetic:PARameter<n>:SPECtrum....1507
MEASurement<m>:SPECtrum:ARIThmetic:PARameter<n>:TYPE....1508
MEASurement<m>:SPECtrum:ATHReshold....1514
MEASurement<m>:SPECtrum:CPOWer:BANDwidth....1512
MEASurement<m>:SPECtrum:CPOWer:CFRequency....1513
MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:LIMit....1517
MEASurement<m>:SPECtrum:LCHeck<n>:LOWer:MARGin....1518
MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:LIMit....1517
MEASurement<m>:SPECtrum:LCHeck<n>:UPPer:MARGin....1518
MEASurement<m>:SPECtrum:LCHeck<n>:VALid....1517
MEASurement<m>:SPECtrum:NDBDown....1513
MEASurement<m>:SPECtrum:OBANdwidth....1513
MEASurement<m>:SPECtrum:PEXCursion....1514
MEASurement<m>:SPECtrum:RESult<n>:COUNt....1514
MEASurement<m>:SPECtrum:RESult<n>:MODE....1515
MEASurement<m>:SSRC....1482
MEASurement<m>:STATistics:HBINs....1535
MEASurement<m>:STATistics:HISTogram....1534
MEASurement<m>:STATistics:MODE....1535
MEASurement<m>:STATistics:RCOunt....1535

MEASurement<m>:STATistics:RESet ... 1536
MEASurement<m>:STATistics:RMEascount ... 1536
MEASurement<m>:STATistics:RTIMe ... 1536
MEASurement<m>:STATistics[:ENABle] ... 1533
MEASurement<m>:TRACk:DATA:HEADer? ... 1545
MEASurement<m>:TRACk:DATA:HEADer? ... 1751
MEASurement<m>:TRACk:DATA:STYPe? ... 1546
MEASurement<m>:TRACk:DATA:STYPe? ... 1751
MEASurement<m>:TRACk:DATA[:VALues]? ... 1546
MEASurement<m>:TRACk:DATA[:VALues]? ... 1752
MEASurement<m>:TRACk[:STATe] ... 1545
MEASurement<m>:TRACk[:STATe] ... 1751
MEASurement<m>:VERTical:AUTO ... 1537
MEASurement<m>:VERTical:CONT ... 1537
MEASurement<m>:VERTical:OFFSet ... 1537
MEASurement<m>:VERTical:SCALe ... 1538
MEASurement<m>[:ENABle] ... 1480
MMEMory:ATTRibute ... 1646
MMEMory:AUTonaming:DATE ... 1667
MMEMory:AUTonaming:DEFaultpath ... 1667
MMEMory:AUTonaming:INDex ... 1667
MMEMory:AUTonaming:PREFix ... 1666
MMEMory:AUTonaming:RESall ... 1667
MMEMory:AUTonaming:RESPath ... 1667
MMEMory:AUTonaming:TEXT ... 1667
MMEMory:AUTonaming:TIME ... 1667
MMEMory:AUTonaming:USERtext ... 1666
MMEMory:CATalog:LENGth? ... 1644
MMEMory:CATalog? ... 1643
MMEMory:CDIRectory ... 1642
MMEMory:COPY ... 1644
MMEMory:DATA ... 1645
MMEMory:DCATalog:LENGth? ... 1642
MMEMory:DCATalog? ... 1641
MMEMory:DELete ... 1645
MMEMory:DRIVes? ... 1641
MMEMory:LOAD:STATe ... 1647
MMEMory:MDIRectory ... 1642
MMEMory:MOVE ... 1644
MMEMory:MSIS ... 1641
MMEMory:NAME ... 1661
MMEMory:RCL ... 1646
MMEMory:RDIRectory ... 1643
MMEMory:SAV ... 1646
MMEMory:STORe:STATe ... 1647
MTESt:ADD ... 1582
MTESt:CEMask ... 1599
MTESt:COLor:BORDer ... 1318
MTESt:COLor:INTerior ... 1319
MTESt:COLor:MATCh ... 1318

MTESt:COLor:UNMatch.... 1318
MTESt:CONDition.... 1584
MTESt:CTYPe.... 1585
MTESt:EYEMask:BOFFset.... 1602
MTESt:EYEMask:HEIGht<m>[:VALue].... 1600
MTESt:EYEMask:HPERiod.... 1601
MTESt:EYEMask:HPOSition.... 1603
MTESt:EYEMask:MSKBottom.... 1601
MTESt:EYEMask:MSKLeft.... 1601
MTESt:EYEMask:MSKRight.... 1601
MTESt:EYEMask:MSKTop.... 1601
MTESt:EYEMask:TBSYmmetric.... 1602
MTESt:EYEMask:TBWidth.... 1602
MTESt:EYEMask:TOFFset.... 1602
MTESt:EYEMask:TYPE.... 1600
MTESt:EYEMask:VPOSition.... 1603
MTESt:EYEMask:WIDTh<m>[:VALue].... 1600
MTESt:FILE:DELete.... 1586
MTESt:FILE:NAME.... 1585
MTESt:FILE:OPEN.... 1586
MTESt:FILE:SAVE.... 1586
MTESt:HIGHlight:INFinite.... 1317
MTESt:HIGHlight:STATe.... 1317
MTESt:HIGHlight:TIME.... 1317
MTESt:LABel.... 1317
MTESt:ONViolation:BEEP.... 1594
MTESt:ONViolation:PRINt.... 1595
MTESt:ONViolation:REPort.... 1596
MTESt:ONViolation:RUNexec.... 1596
MTESt:ONViolation:SAVewaveform.... 1595
MTESt:ONViolation:STOP.... 1595
MTESt:ONViolation:TRIGgerout.... 1596
MTESt:REFWfm.... 1592
MTESt:REMove.... 1583
MTESt:REName.... 1317
MTESt:RESult:COUNt:FAILures?.... 1598
MTESt:RESult:COUNt:FSLices?.... 2418
MTESt:RESult:COUNt:FWAVeforms?.... 1598
MTESt:RESult:COUNt:REMaining?.... 1598
MTESt:RESult:COUNt:SLICes?.... 2418
MTESt:RESult:COUNt:WAVeforms?.... 1597
MTESt:RESult:FRATe?.... 1599
MTESt:RESult:STATe?.... 1597
MTESt:RESult:VIOLation<m>:COUNt?.... 2418
MTESt:RESult:VIOLation<m>:STARt?.... 2419
MTESt:RESult:VIOLation<m>:STOP?.... 2419
MTESt:RESult:VIOLation<m>[:SLICe]?.... 2419
MTESt:RESult[:RESult]?.... 1597
MTESt:RST.... 1583
MTESt:SBITnumber?.... 2577

MTESt:SEGMent:ADD....1587
MTESt:SEGMent:CLEar....1588
MTESt:SEGMent:COUNt?....1587
MTESt:SEGMent:INSert....1587
MTESt:SEGMent:POINt:ADD....1589
MTESt:SEGMent:POINt:COUNt?....1589
MTESt:SEGMent:POINt:INSert....1589
MTESt:SEGMent:POINt:REMove....1589
MTESt:SEGMent:POINt:X....1590
MTESt:SEGMent:POINt:Y....1590
MTESt:SEGMent:REGion....1588
MTESt:SEGMent:REMove....1588
MTESt:SEGMent:RESCale:RECalculate....1590
MTESt:SEGMent:RESCale:XFACtor....1591
MTESt:SEGMent:RESCale:XOFFset....1591
MTESt:SEGMent:RESCale:YFACtor....1591
MTESt:SEGMent:RESCale:YOFFset....1592
MTESt:SEGMent:STATe....1587
MTESt:SHOW....1583
MTESt:SOURce....1584
MTESt:TOLerance....1584
MTESt:WFMLupdate....1593
MTESt:WFMRescale:XWIDth....1593
MTESt:WFMRescale:YPOSition....1594
MTESt:WFMRescale:YSTRetch....1594
MTESt:WFMRescale:YWIDth....1593
MTESt[:STATe]....1583
PGENerator:BITRate....2456
PGENerator:ENABle....2456
PGENerator:FILE:OPEN....2456
PGENerator:FILE[:NAME]....2456
PGENerator:HLEVel....2457
PGENerator:PRESet....2457
PGENerator:RUNMode....2457
PGENerator:RUNSingle....2457
POWer:DESKew:CURRent....2521
POWer:DESKew:EXECute....2521
POWer:DESKew:RESet....2521
POWer:DESKew:TIME?....2521
POWer:DESKew:UDPReset....2521
POWer:DONRes:AUTO....2535
POWer:DONRes:AVG....2535
POWer:DONRes:EXECute....2536
POWer:DONRes:GATE<m>:STARt....2536
POWer:DONRes:GATE<m>:STOP....2536
POWer:DONRes:REPort:ADD....2536
POWer:DONRes:RESult:RESistance?....2536
POWer:EFFiciency:AUTO....2546
POWer:EFFiciency:EXECute....2547
POWer:EFFiciency:REPort:ADD....2547

POWer:EFFiciency:RESult<m>:ACTual?.....2547
POWer:EFFiciency:RESult<m>:AVG?.....2547
POWer:EFFiciency:RESult<m>:EVTCount?.....2547
POWer:EFFiciency:RESult<m>:NPEak?.....2547
POWer:EFFiciency:RESult<m>:PPEak?.....2547
POWer:EFFiciency:RESult<m>:RMS?.....2547
POWer:EFFiciency:RESult<m>:STDDev?.....2547
POWer:EFFiciency:RESult<m>:WFMCount?.....2547
POWer:ENABle.....2520
POWer:HARMonics:AUTO.....2531
POWer:HARMonics:DOFR.....2531
POWer:HARMonics:ENFR.....2531
POWer:HARMonics:EVAL.....2532
POWer:HARMonics:EXECute.....2532
POWer:HARMonics:MIFR.....2532
POWer:HARMonics:REPort:ADD.....2532
POWer:HARMonics:RESult<m>:FREQuency<n>:VALue?.....2532
POWer:HARMonics:RESult<m>:MAXValue<n>:VALue?.....2532
POWer:HARMonics:RESult<m>:STDinuse?.....2532
POWer:HARMonics:RESult<m>:STDValue<n>:VALue?.....2532
POWer:HARMonics:RESult<m>:VALue<n>:VALue?.....2532
POWer:HARMonics:STAN.....2533
POWer:HARMonics:VOLT.....2533
POWer:INRush:ADD.....2529
POWer:INRush:COUNt?.....2529
POWer:INRush:EXECute.....2530
POWer:INRush:GATE<m>:STARt.....2530
POWer:INRush:GATE<m>:STOP.....2530
POWer:INRush:GATE<m>:VALue.....2530
POWer:INRush:INSert.....2529
POWer:INRush:MAXCurrent.....2530
POWer:INRush:REMove.....2529
POWer:INRush:REPort:ADD.....2531
POWer:INRush:TRIGger.....2530
POWer:MODulation:AUTO.....2534
POWer:MODulation:DHIStogram.....2534
POWer:MODulation:EXECute.....2534
POWer:MODulation:REPort:ADD.....2534
POWer:MODulation:RESult:ACTual?.....2534
POWer:MODulation:RESult:AVG?.....2534
POWer:MODulation:RESult:EVTCount?.....2534
POWer:MODulation:RESult:NPEak?.....2534
POWer:MODulation:RESult:PPEak?.....2534
POWer:MODulation:RESult:RMS?.....2534
POWer:MODulation:RESult:STDDev?.....2534
POWer:MODulation:RESult:WFMCount?.....2534
POWer:MODulation:SOURce.....2535
POWer:MODulation:TYPE.....2535
POWer:ONOFf:ATOFf.....2542
POWer:ONOFf:ATON.....2542

POWer:ONOFf:DSOFf....2542
POWer:ONOFf:DSON....2542
POWer:ONOFf:DTOFf....2542
POWer:ONOFf:DTON....2542
POWer:ONOFf:EXECute....2542
POWer:ONOFf:INPut....2542
POWer:ONOFf:REPort:ADD....2542
POWer:ONOFf:RESult:TOFF?....2543
POWer:ONOFf:RESult:TON?....2543
POWer:ONOFf:TIME....2543
POWer:ONOFf:TYPE....2543
POWer:QUALity:AUTO....2527
POWer:QUALity:EXECute....2527
POWer:QUALity:FCUS....2528
POWer:QUALity:FREQ....2527
POWer:QUALity:REPort:ADD....2528
POWer:QUALity:RESult:CURRent:CREStfactor?....2528
POWer:QUALity:RESult:CURRent:FREQuency?....2528
POWer:QUALity:RESult:CURRent:PEAK?....2528
POWer:QUALity:RESult:CURRent:RMS?....2528
POWer:QUALity:RESult:POWer:APParent?....2528
POWer:QUALity:RESult:POWer:PFACtor?....2528
POWer:QUALity:RESult:POWer:PHASe?....2528
POWer:QUALity:RESult:POWer:REACtive?....2528
POWer:QUALity:RESult:POWer:REALpower?....2528
POWer:QUALity:RESult:VOLTage:CREStfactor?....2528
POWer:QUALity:RESult:VOLTage:FREQuency?....2528
POWer:QUALity:RESult:VOLTage:PEAK?....2528
POWer:QUALity:RESult:VOLTage:RMS?....2528
POWer:REPort:CONTent:HSETup....2522
POWer:REPort:CONTent:MSETup....2522
POWer:REPort:CONTent:MSIGnal....2522
POWer:REPort:CONTent:RESU....2522
POWer:REPort:CONTent:SETTings....2522
POWer:REPort:CONTent:TITLe....2523
POWer:REPort:CONTent:TSETup....2522
POWer:REPort:CONTent:VSETup....2522
POWer:REPort:DESCription....2523
POWer:REPort:DUT....2523
POWer:REPort:FILE:DELete....2524
POWer:REPort:FILE:NAME....2524
POWer:REPort:FILE:NEW....2524
POWer:REPort:FILE:SAVE....2524
POWer:REPort:FONT:COLO....2523
POWer:REPort:FONT:FAMI....2523
POWer:REPort:FONT:SIZE....2523
POWer:REPort:INVert....2525
POWer:REPort:LOGO....2523
POWer:REPort:PAPersize....2524
POWer:REPort:SITE....2523

POWer:REPort:TEMPerature.....2523
POWer:REPort:TEST:ADD.....2524
POWer:REPort:TEST:COMMent.....2526
POWer:REPort:TEST:COUNt.....2526
POWer:REPort:TEST:DIRectory.....2526
POWer:REPort:TEST:DSEA.....2525
POWer:REPort:TEST:INSert.....2524
POWer:REPort:TEST:ISE.....2525
POWer:REPort:TEST:LSENd?.....2527
POWer:REPort:TEST:REMove.....2524
POWer:REPort:TEST:RSE.....2525
POWer:REPort:TEST:SEA.....2525
POWer:REPort:USER.....2523
POWer:RIPPle:AUToscale.....2549
POWer:RIPPle:CURRent.....2549
POWer:RIPPle:EXECute.....2549
POWer:RIPPle:FREQuency.....2549
POWer:RIPPle:REPort:ADD.....2550
POWer:RIPPle:RESult:FREQuency:AVG?.....2550
POWer:RIPPle:RESult:FREQuency:EVTCount?.....2550
POWer:RIPPle:RESult:FREQuency:NPEak?.....2550
POWer:RIPPle:RESult:FREQuency:PPEak?.....2550
POWer:RIPPle:RESult:FREQuency:RMS?.....2550
POWer:RIPPle:RESult:FREQuency:STDDev?.....2550
POWer:RIPPle:RESult:FREQuency:WFMCount?.....2550
POWer:RIPPle:RESult:FREQuency[:ACTual]?.....2550
POWer:RIPPle:RESult:MAXimum:AVG?.....2550
POWer:RIPPle:RESult:MAXimum:EVTCount?.....2550
POWer:RIPPle:RESult:MAXimum:NPEak?.....2550
POWer:RIPPle:RESult:MAXimum:PPEak?.....2550
POWer:RIPPle:RESult:MAXimum:RMS?.....2550
POWer:RIPPle:RESult:MAXimum:STDDev?.....2550
POWer:RIPPle:RESult:MAXimum:WFMCount?.....2550
POWer:RIPPle:RESult:MAXimum[:ACTual]?.....2550
POWer:RIPPle:RESult:MINimum:AVG?.....2551
POWer:RIPPle:RESult:MINimum:EVTCount?.....2551
POWer:RIPPle:RESult:MINimum:NPEak?.....2551
POWer:RIPPle:RESult:MINimum:PPEak?.....2551
POWer:RIPPle:RESult:MINimum:RMS?.....2551
POWer:RIPPle:RESult:MINimum:STDDev?.....2551
POWer:RIPPle:RESult:MINimum:WFMCount?.....2551
POWer:RIPPle:RESult:MINimum[:ACTual]?.....2551
POWer:RIPPle:RESult:NDCYcle:AVG?.....2551
POWer:RIPPle:RESult:NDCYcle:EVTCount?.....2551
POWer:RIPPle:RESult:NDCYcle:NPEak?.....2551
POWer:RIPPle:RESult:NDCYcle:PPEak?.....2551
POWer:RIPPle:RESult:NDCYcle:RMS?.....2551
POWer:RIPPle:RESult:NDCYcle:STDDev?.....2552
POWer:RIPPle:RESult:NDCYcle:WFMCount?.....2552
POWer:RIPPle:RESult:NDCYcle[:ACTual]?.....2551

POWer:RIPPle:RESult:PDCYcle:AVG?....2552
POWer:RIPPle:RESult:PDCYcle:EVTCount?....2552
POWer:RIPPle:RESult:PDCYcle:NPEak?....2552
POWer:RIPPle:RESult:PDCYcle:PPEak?....2552
POWer:RIPPle:RESult:PDCYcle:RMS?....2552
POWer:RIPPle:RESult:PDCYcle:STDDev?....2552
POWer:RIPPle:RESult:PDCYcle:WFMCount?....2552
POWer:RIPPle:RESult:PDCYcle[:ACTual]?....2552
POWer:RIPPle:RESult:PDEL:AVG?....2553
POWer:RIPPle:RESult:PDEL:EVTCount?....2553
POWer:RIPPle:RESult:PDEL:NPEak?....2553
POWer:RIPPle:RESult:PDEL:PPEak?....2553
POWer:RIPPle:RESult:PDEL:RMS?....2553
POWer:RIPPle:RESult:PDEL:STDDev?....2553
POWer:RIPPle:RESult:PDEL:WFMCount?....2553
POWer:RIPPle:RESult:PDEL[:ACTual]?....2553
POWer:RIPPle:RESult:PERiod:AVG?....2553
POWer:RIPPle:RESult:PERiod:EVTCount?....2553
POWer:RIPPle:RESult:PERiod:NPEak?....2553
POWer:RIPPle:RESult:PERiod:PPEak?....2553
POWer:RIPPle:RESult:PERiod:RMS?....2553
POWer:RIPPle:RESult:PERiod:STDDev?....2553
POWer:RIPPle:RESult:PERiod:WFMCount?....2553
POWer:RIPPle:RESult:PERiod[:ACTual]?....2553
POWer:RIPPle:RESult:STDDev:AVG?....2554
POWer:RIPPle:RESult:STDDev:EVTCount?....2554
POWer:RIPPle:RESult:STDDev:NPEak?....2554
POWer:RIPPle:RESult:STDDev:PPEak?....2554
POWer:RIPPle:RESult:STDDev:RMS?....2554
POWer:RIPPle:RESult:STDDev:STDDev?....2554
POWer:RIPPle:RESult:STDDev:WFMCount?....2554
POWer:RIPPle:RESult:STDDev[:ACTual]?....2554
POWer:SLEWrate:AUTO....2537
POWer:SLEWrate:AVGDeriv....2537
POWer:SLEWrate:EXECute....2537
POWer:SLEWrate:GATE:STARt....2538
POWer:SLEWrate:GATE:STOP....2538
POWer:SLEWrate:REPort:ADD....2538
POWer:SLEWrate:RESult:ACTual?....2538
POWer:SLEWrate:RESult:AVG?....2538
POWer:SLEWrate:RESult:EVTCount?....2538
POWer:SLEWrate:RESult:NPEak?....2538
POWer:SLEWrate:RESult:PPEak?....2538
POWer:SLEWrate:RESult:RMS?....2538
POWer:SLEWrate:RESult:STDDev?....2539
POWer:SLEWrate:RESult:WFMCount?....2539
POWer:SLEWrate:SOURce....2537
POWer:SOA:EXECute....2539
POWer:SOA:LINear:ADD....2539
POWer:SOA:LINear:COUNt?....2540

POWer:SOA:LINear:INSert....2540
POWer:SOA:LINear:POINt<m>:CURRent....2540
POWer:SOA:LINear:POINt<m>:VOLTage....2540
POWer:SOA:LINear:REMove....2540
POWer:SOA:LOGarithmic:ADD....2539
POWer:SOA:LOGarithmic:COUNt?....2540
POWer:SOA:LOGarithmic:INSert....2540
POWer:SOA:LOGarithmic:POINt<m>:CURRent....2540
POWer:SOA:LOGarithmic:POINt<m>:VOLTage....2540
POWer:SOA:LOGarithmic:REMove....2540
POWer:SOA:MASK....2541
POWer:SOA:REPort:ADD....2541
POWer:SOA:SCALe....2541
POWer:SOA:SWITch....2541
POWer:SOURce:CURRent<1..2>....2520
POWer:SOURce:VOLTage<1..4>....2520
POWer:SPECtrum:AUToscale....2557
POWer:SPECtrum:EXECute....2557
POWer:SPECtrum:FREQuency....2557
POWer:SPECtrum:RCOunt?....2558
POWer:SPECtrum:REPort:ADD....2558
POWer:SPECtrum:RESult<m>:FREQuency?....2558
POWer:SPECtrum:RESult<m>:LEVel?....2558
POWer:SWITching:AUTO....2544
POWer:SWITching:COND....2545
POWer:SWITching:EXECute....2544
POWer:SWITching:GATE:COND:STARt....2545
POWer:SWITching:GATE:COND:STOP....2545
POWer:SWITching:GATE:NCON:STARt....2545
POWer:SWITching:GATE:TOFF:STARt....2545
POWer:SWITching:GATE:TOFF:STOP....2545
POWer:SWITching:GATE:TON:STARt....2545
POWer:SWITching:GATE:TON:STOP....2545
POWer:SWITching:NCON....2545
POWer:SWITching:REPort:ADD....2544
POWer:SWITching:RESult:ENERgy:ACTual?....2545
POWer:SWITching:RESult:ENERgy:AVG?....2545
POWer:SWITching:RESult:ENERgy:EVTCount?....2545
POWer:SWITching:RESult:ENERgy:NPEak?....2545
POWer:SWITching:RESult:ENERgy:PPEak?....2545
POWer:SWITching:RESult:ENERgy:RMS?....2545
POWer:SWITching:RESult:ENERgy:STDDev?....2545
POWer:SWITching:RESult:ENERgy:WFMCount?....2545
POWer:SWITching:RESult:POWer:ACTual?....2546
POWer:SWITching:RESult:POWer:AVG?....2546
POWer:SWITching:RESult:POWer:EVTCount?....2546
POWer:SWITching:RESult:POWer:NPEak?....2546
POWer:SWITching:RESult:POWer:PPEak?....2546
POWer:SWITching:RESult:POWer:RMS?....2546
POWer:SWITching:RESult:POWer:STDDev?....2546

POWer:SWITching:RESult:POWer:WFMCount?....2546
POWer:SWITching:SWIFrequency....2544
POWer:SWITching:SWIT....2545
POWer:SWITching:TOFF....2545
POWer:SWITching:TON....2545
POWer:SWITching:TOTal....2545
POWer:TRANsient:AUToscale....2555
POWer:TRANsient:EXECute....2555
POWer:TRANsient:FREQuency....2555
POWer:TRANsient:HYSTeresis....2555
POWer:TRANsient:INPut....2555
POWer:TRANsient:REPort:ADD....2556
POWer:TRANsient:RESult[:ACTual]?....2556
POWer:TRANsient:SIGHigh....2556
POWer:TRANsient:SIGLow....2556
POWer:TRANsient:TRGChannel....2556
POWer:TRANsient:TRGLevel....2557
POWer:TRANsient:TRGSlope....2557
PROBe<m>:DEEMbedding:BANDwidth....1359
PROBe<m>:DEEMbedding:BANDwidth....2566
PROBe<m>:DEEMbedding:LOAD:IMPedance....1361
PROBe<m>:DEEMbedding:MODE....1360
PROBe<m>:DEEMbedding:PMODe....1358
PROBe<m>:DEEMbedding:REMProbeload....1360
PROBe<m>:DEEMbedding:TIPModule:CSPFile....1359
PROBe<m>:DEEMbedding:TIPModule:ZMA<n>:SUBModule....1359
PROBe<m>:DEEMbedding:TIPModule:ZMA<n>:SUBModule....2566
PROBe<m>:DEEMbedding:TIPModule[:SELect]....1358
PROBe<m>:DEEMbedding:TIPModule[:SELect]....2566
PROBe<m>:ID:PARTnumber?....1364
PROBe<m>:ID:PRDate?....1363
PROBe<m>:ID:SRNumber?....1364
PROBe<m>:ID:SWVersion?....1363
PROBe<m>:PMETer:RESults:COMMon?....1349
PROBe<m>:PMETer:RESults:DIFFerential?....1349
PROBe<m>:PMETer:RESults:NEGative?....1350
PROBe<m>:PMETer:RESults:POSitive?....1350
PROBe<m>:PMETer:RESults:SINGle?....1349
PROBe<m>:PMETer:VISibility....1348
PROBe<m>:SETup:ACCoupling....1353
PROBe<m>:SETup:ADAPter?....1365
PROBe<m>:SETup:ADVanced:AUDioverload....1354
PROBe<m>:SETup:ADVanced:FILTer....1354
PROBe<m>:SETup:ADVanced:PMToffset....1353
PROBe<m>:SETup:ADVanced:RANGe....1353
PROBe<m>:SETup:ATTenuation:DEFProbe....1361
PROBe<m>:SETup:ATTenuation:MANual....1351
PROBe<m>:SETup:ATTenuation:MODE....1351
PROBe<m>:SETup:ATTenuation:TDEFprobe....1365
PROBe<m>:SETup:ATTenuation:UNIT....1351

PROBe<m>:SETup:ATTenuation[:AUTO]?....1345
PROBe<m>:SETup:BANDwidth?....1345
PROBe<m>:SETup:CAPacitance?....1364
PROBe<m>:SETup:CMOFfset....1352
PROBe<m>:SETup:CMOFfset....1355
PROBe<m>:SETup:DEGauss....1362
PROBe<m>:SETup:DISPlaydiff....1348
PROBe<m>:SETup:DMOFfset....1355
PROBe<m>:SETup:GAIN:MANual....1351
PROBe<m>:SETup:IMPedance?....1364
PROBe<m>:SETup:MODE....1347
PROBe<m>:SETup:NAME?....1344
PROBe<m>:SETup:NOFFset....1356
PROBe<m>:SETup:OFFSet:AZERo....1345
PROBe<m>:SETup:OFFSet:STPRobe....1362
PROBe<m>:SETup:OFFSet:TOMean....1362
PROBe<m>:SETup:OFFSet:USEautozero....1346
PROBe<m>:SETup:OFFSet:ZADJust....1363
PROBe<m>:SETup:POFFset....1356
PROBe<m>:SETup:PRMode....1354
PROBe<m>:SETup:STATe?....1344
PROBe<m>:SETup:TERM:ADJust....1357
PROBe<m>:SETup:TERM:MEASure?....1357
PROBe<m>:SETup:TERM:MODE....1357
PROBe<m>:SETup:TERM:STATe....1356
PROBe<m>:SETup:TYPE?....1344
PROBe<m>:SETup:ZAXV....1352
PROBe<m>:SKEState....1385
PSRC:CLOCkmode....2465
PSRC:DUTYcycle....2465
PSRC:OUTPutlow....2464
PSRC:REPRate....2465
PSRC:RST....2465
PSRC[:STAT]....2464
REFCurve<m>:CLEar....1448
REFCurve<m>:DATA:HEADer?....1453
REFCurve<m>:DATA:STYPe?....1453
REFCurve<m>:DATA[:VALues]?....1454
REFCurve<m>:DELete....1448
REFCurve<m>:HMODe....1451
REFCurve<m>:MULTichannel:IMPort....1455
REFCurve<m>:MULTichannel:NAME....1455
REFCurve<m>:MULTichannel:OPEN....1455
REFCurve<m>:NAME....1447
REFCurve<m>:OPEN....1447
REFCurve<m>:POSition....1450
REFCurve<m>:RESCale:HORizontal:FACTor....1452
REFCurve<m>:RESCale:HORizontal:OFFSet....1452
REFCurve<m>:RESCale:HORizontal:POSition....1453
REFCurve<m>:RESCale:HORizontal:STATe....1452

REFCurve<m>:RESCale:VERTical:FACTor.... 1450
REFCurve<m>:RESCale:VERTical:OFFSet.... 1451
REFCurve<m>:RESCale:VERTical:STATe.... 1450
REFCurve<m>:RESTore.... 1449
REFCurve<m>:SAVE.... 1448
REFCurve<m>:SCALe.... 1449
REFCurve<m>:SOURce.... 1446
REFCurve<m>:STATe.... 1447
REFCurve<m>:TOORignal.... 1450
REFCurve<m>:UPDate.... 1448
REFCurve<m>:VMODe.... 1449
REFLevel<m>:ABSolute:BDIStance.... 1558
REFLevel<m>:ABSolute:BITube.... 1562
REFLevel<m>:ABSolute:BOTube.... 1563
REFLevel<m>:ABSolute:HIGH.... 1556
REFLevel<m>:ABSolute:LLEVel.... 1559
REFLevel<m>:ABSolute:LOW.... 1557
REFLevel<m>:ABSolute:MLEVel.... 1558
REFLevel<m>:ABSolute:TDIStance.... 1557
REFLevel<m>:ABSolute:TITube.... 1562
REFLevel<m>:ABSolute:TOTube.... 1562
REFLevel<m>:ABSolute:ULEVel.... 1558
REFLevel<m>:AUTO:COUNt.... 1556
REFLevel<m>:AUTO:MODE.... 1554
REFLevel<m>:AUTO[:STATe].... 1555
REFLevel<m>:LDETection.... 1552
REFLevel<m>:LMODe.... 1553
REFLevel<m>:RELative:HYSTeresis.... 1561
REFLevel<m>:RELative:ITUBe.... 1561
REFLevel<m>:RELative:LOWer.... 1560
REFLevel<m>:RELative:MIDDle.... 1560
REFLevel<m>:RELative:MODE.... 1553
REFLevel<m>:RELative:OTUBe.... 1561
REFLevel<m>:RELative:UPPer.... 1559
REFLevel<m>:USRLevel.... 1554
REPort:COMMent.... 1666
REPort:FILE:NAME.... 1666
REPort:FILE:SAVE.... 1666
REPort:LANGuage.... 1665
REPort:LOGO.... 1665
REPort:LOGType.... 1665
REPort:PAPersize.... 1665
REPort:USER.... 1666
RUN.... 1325
RUNContinous.... 1325
RUNSingle.... 1325
SAVeset:CONFig:PREView.... 1648
SAVeset:ONEFile:NAME.... 1648
SAVeset:ONEFile:OPEN.... 1648
SAVeset:ONEFile:SAVE.... 1648

SAVeset:ONEFile:USERpref....1648
SEARch:ADD....1604
SEARch:ALL....1605
SEARch:CLEar....1604
SEARch:GATE:ABSolute:STARt....1632
SEARch:GATE:ABSolute:STOP....1632
SEARch:GATE:MODE....1632
SEARch:GATE:RELative:STARt....1633
SEARch:GATE:RELative:STOP....1633
SEARch:GATE:SHOW....1632
SEARch:GATE:ZCOupling....1633
SEARch:GATE:ZDIagram....1634
SEARch:GATE[:STATe]....1631
SEARch:ONLine....1605
SEARch:REMove....1605
SEARch:RESDiagram:HORZ:ABSolute:POSition....1634
SEARch:RESDiagram:HORZ:ABSolute:SPAN....1635
SEARch:RESDiagram:HORZ:MODE....1635
SEARch:RESDiagram:HORZ:RELative:POSition....1635
SEARch:RESDiagram:HORZ:RELative:SPAN....1636
SEARch:RESDiagram:SHOW....1636
SEARch:RESDiagram:VERT:ABSolute:POSition....1636
SEARch:RESDiagram:VERT:ABSolute:SPAN....1636
SEARch:RESDiagram:VERT:MODE....1637
SEARch:RESDiagram:VERT:RELative:POSition....1637
SEARch:RESDiagram:VERT:RELative:SPAN....1637
SEARch:RESult:ARINc:WCOunt?....1998
SEARch:RESult:ARINc:WORD<m>:DATA?....1997
SEARch:RESult:ARINc:WORD<m>:LABel?....1996
SEARch:RESult:ARINc:WORD<m>:PATTern?....1996
SEARch:RESult:ARINc:WORD<m>:SDI?....1998
SEARch:RESult:ARINc:WORD<m>:SSM?....1997
SEARch:RESult:ARINc:WORD<m>:STARt?....1999
SEARch:RESult:ARINc:WORD<m>:STATe?....1998
SEARch:RESult:ARINc:WORD<m>:STOP?....1998
SEARch:RESult:ARINc:WORD<m>:SYMBol?....1997
SEARch:RESult:CAN:FCOunt?....1826
SEARch:RESult:CAN:FDATa:FRAMe<m>:SCValue?....1830
SEARch:RESult:CAN:FDATa:FRAMe<m>:STANdard?....1830
SEARch:RESult:CAN:FRAMe<m>:ACKState?....1828
SEARch:RESult:CAN:FRAMe<m>:ACKValue?....1826
SEARch:RESult:CAN:FRAMe<m>:BSEPosition?....1827
SEARch:RESult:CAN:FRAMe<m>:BYTE<n>:STATe?....1827
SEARch:RESult:CAN:FRAMe<m>:BYTE<n>:VALue?....1827
SEARch:RESult:CAN:FRAMe<m>:CSSTate?....1828
SEARch:RESult:CAN:FRAMe<m>:CSValue?....1828
SEARch:RESult:CAN:FRAMe<m>:DATA?....1828
SEARch:RESult:CAN:FRAMe<m>:DLCState?....1828
SEARch:RESult:CAN:FRAMe<m>:DLCValue?....1828
SEARch:RESult:CAN:FRAMe<m>:FERCause?....1829

SEARch:RESult:CAN:FRAMe<m>:IDSTate? 1828
SEARch:RESult:CAN:FRAMe<m>:IDTYpe? 1829
SEARch:RESult:CAN:FRAMe<m>:IDValue? 1829
SEARch:RESult:CAN:FRAMe<m>:NDBYtes? 1822
SEARch:RESult:CAN:FRAMe<m>:SDATa? 1834
SEARch:RESult:CAN:FRAMe<m>:STARt? 1830
SEARch:RESult:CAN:FRAMe<m>:STATus? 1830
SEARch:RESult:CAN:FRAMe<m>:STOP? 1831
SEARch:RESult:CAN:FRAMe<m>:SYMBol? 1831
SEARch:RESult:CAN:FRAMe<m>:TYPE? 1831
SEARch:RESult:CMSB<m>:FCOunt? 1789
SEARch:RESult:CMSB<m>:FRAMe<n>:CCOunt? 1789
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:NAME? 1790
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STARt? 1790
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STATe? 1791
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:STOP? 1791
SEARch:RESult:CMSB<m>:FRAMe<n>:CELL<o>:VALue? 1792
SEARch:RESult:CMSB<m>:FRAMe<n>:CONE? 1792
SEARch:RESult:CMSB<m>:FRAMe<n>:CTHRee? 1792
SEARch:RESult:CMSB<m>:FRAMe<n>:CTWO? 1793
SEARch:RESult:CMSB<m>:FRAMe<n>:STARt? 1793
SEARch:RESult:CMSB<m>:FRAMe<n>:STATe? 1793
SEARch:RESult:CMSB<m>:FRAMe<n>:STOP? 1794
SEARch:RESult:CMSB<m>:FRAMe<n>:TYPE? 1794
SEARch:RESult:CXPI:FCOunt? 1954
SEARch:RESult:CXPI:FRAMe<m>:DATA? 1954
SEARch:RESult:CXPI:FRAMe<m>:DLCV? 1954
SEARch:RESult:CXPI:FRAMe<m>:STARt? 1955
SEARch:RESult:CXPI:FRAMe<m>:STATe? 1955
SEARch:RESult:CXPI:FRAMe<m>:STOP? 1955
SEARch:RESult:CXPI:FRAMe<m>:TYPE? 1955
SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:STATus? 1956
SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:TYPE? 1956
SEARch:RESult:CXPI:FRAMe<m>:WORD<n>:VALue? 1956
SEARch:RESult:DDR:FCOunt? 2398
SEARch:RESult:DDR:FRAMe<m>:FLD<n>:FVAL? 2398
SEARch:RESult:DDR:FRAMe<m>:FLD<n>:STARt? 2399
SEARch:RESult:DDR:FRAMe<m>:FLD<n>:STATus? 2399
SEARch:RESult:DDR:FRAMe<m>:FLD<n>:STOP? 2400
SEARch:RESult:DDR:FRAMe<m>:FLD<n>:VAL? 2400
SEARch:RESult:DDR:FRAMe<m>:NWRDs? 2400
SEARch:RESult:DDR:FRAMe<m>:STARt? 2401
SEARch:RESult:DDR:FRAMe<m>:STATe? 2401
SEARch:RESult:DDR:FRAMe<m>:STOP? 2401
SEARch:RESult:DDR:FRAMe<m>:TYPE? 2402
SEARch:RESult:DPHY:FCOunt? 2269
SEARch:RESult:DPHY:FRAMe<m>:CS? 2269
SEARch:RESult:DPHY:FRAMe<m>:DATA? 2270
SEARch:RESult:DPHY:FRAMe<m>:DTName? 2270
SEARch:RESult:DPHY:FRAMe<m>:DTYPe? 2270

SEARch:RESult:DPHY:FRAMe<m>:ECC? 2271
SEARch:RESult:DPHY:FRAMe<m>:PACKet<n>:IDX? 2271
SEARch:RESult:DPHY:FRAMe<m>:PACKet<n>:VALue? 2271
SEARch:RESult:DPHY:FRAMe<m>:STARt? 2272
SEARch:RESult:DPHY:FRAMe<m>:STATe? 2272
SEARch:RESult:DPHY:FRAMe<m>:STOP? 2272
SEARch:RESult:DPHY:FRAMe<m>:TYPE? 2273
SEARch:RESult:DPHY:FRAMe<m>:VCHannel? 2273
SEARch:RESult:EBTB:SCOunt? 2345
SEARch:RESult:EBTB:SYMBol<m>:DATA? 2345
SEARch:RESult:EBTB:SYMBol<m>:STARt? 2345
SEARch:RESult:EBTB:SYMBol<m>:STATus? 2345
SEARch:RESult:EBTB:SYMBol<m>:STOP? 2346
SEARch:RESult:ETHernet:WCOunt? 2035
SEARch:RESult:ETHernet:WORD<m>:BYTE<n>:VALue? 2039
SEARch:RESult:ETHernet:WORD<m>:CRC? 2038
SEARch:RESult:ETHernet:WORD<m>:DATA? 2038
SEARch:RESult:ETHernet:WORD<m>:DESTaddress? 2037
SEARch:RESult:ETHernet:WORD<m>:DSYMbol? 2039
SEARch:RESult:ETHernet:WORD<m>:FTYPe? 2038
SEARch:RESult:ETHernet:WORD<m>:SRCaddress? 2037
SEARch:RESult:ETHernet:WORD<m>:SSYMbol? 2039
SEARch:RESult:ETHernet:WORD<m>:STARt? 2036
SEARch:RESult:ETHernet:WORD<m>:STATe? 2036
SEARch:RESult:ETHernet:WORD<m>:STOP? 2037
SEARch:RESult:ETHernet:WORD<m>:TYPE? 2037
SEARch:RESult:FLXRay:FCOunt? 1894
SEARch:RESult:FLXRay:FRAMe<m>:ADID? 1894
SEARch:RESult:FLXRay:FRAMe<m>:CSSTate? 1894
SEARch:RESult:FLXRay:FRAMe<m>:CSValue? 1895
SEARch:RESult:FLXRay:FRAMe<m>:CYCount? 1895
SEARch:RESult:FLXRay:FRAMe<m>:DATA? 1895
SEARch:RESult:FLXRay:FRAMe<m>:FCSTate? 1896
SEARch:RESult:FLXRay:FRAMe<m>:FCValue? 1896
SEARch:RESult:FLXRay:FRAMe<m>:FLAGs? 1896
SEARch:RESult:FLXRay:FRAMe<m>:PAYLength? 1897
SEARch:RESult:FLXRay:FRAMe<m>:STARt? 1898
SEARch:RESult:FLXRay:FRAMe<m>:STATus? 1897
SEARch:RESult:FLXRay:FRAMe<m>:STOP? 1898
SEARch:RESult:FLXRay:FRAMe<m>:SYMBol? 1898
SEARch:RESult:FLXRay:FRAMe<m>:TYPE? 1898
SEARch:RESult:HBTO:FCOunt? 2111
SEARch:RESult:HBTO:FRAMe<m>:CRC? 2114
SEARch:RESult:HBTO:FRAMe<m>:DATA? 2113
SEARch:RESult:HBTO:FRAMe<m>:DESTaddress? 2113
SEARch:RESult:HBTO:FRAMe<m>:DSYMbol? 2115
SEARch:RESult:HBTO:FRAMe<m>:FTYPe? 2114
SEARch:RESult:HBTO:FRAMe<m>:NUMWords? 2114
SEARch:RESult:HBTO:FRAMe<m>:SRCaddress? 2113
SEARch:RESult:HBTO:FRAMe<m>:SSYMbol? 2115

SEARch:RESult:HBTO:FRAMe<m>:STARt?....2112
SEARch:RESult:HBTO:FRAMe<m>:STATe?....2112
SEARch:RESult:HBTO:FRAMe<m>:STOP?....2112
SEARch:RESult:HBTO:FRAMe<m>:TYPE?....2111
SEARch:RESult:HBTO:FRAMe<m>:WORD<n>:TYPE?....2115
SEARch:RESult:HBTO:FRAMe<m>:WORD<n>:VALue?....2116
SEARch:RESult:I2C:FCOunt?....1702
SEARch:RESult:I2C:FRAMe<m>:AACCess?....1703
SEARch:RESult:I2C:FRAMe<m>:ACCess?....1703
SEARch:RESult:I2C:FRAMe<m>:ACOMplete?....1703
SEARch:RESult:I2C:FRAMe<m>:ADBStart?....1704
SEARch:RESult:I2C:FRAMe<m>:ADDRess?....1704
SEARch:RESult:I2C:FRAMe<m>:ADEVice?....1704
SEARch:RESult:I2C:FRAMe<m>:AMODe?....1704
SEARch:RESult:I2C:FRAMe<m>:ASTart?....1705
SEARch:RESult:I2C:FRAMe<m>:BCOunt?....1706
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:ACCess?....1706
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:ACKStart?....1706
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:COMPlete?....1707
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:STARt?....1707
SEARch:RESult:I2C:FRAMe<m>:BYTE<n>:VALue?....1707
SEARch:RESult:I2C:FRAMe<m>:DATA?....1705
SEARch:RESult:I2C:FRAMe<m>:RWBStart?....1705
SEARch:RESult:I2C:FRAMe<m>:STARt?....1702
SEARch:RESult:I2C:FRAMe<m>:STATus?....1702
SEARch:RESult:I2C:FRAMe<m>:STOP?....1702
SEARch:RESult:I2C:FRAMe<m>:SYMBol?....1706
SEARch:RESult:LIMit....1638
SEARch:RESult:LIN:FCOunt?....1860
SEARch:RESult:LIN:FRAMe<m>:BYTE<n>:STATe?....1864
SEARch:RESult:LIN:FRAMe<m>:BYTE<n>:VALue?....1865
SEARch:RESult:LIN:FRAMe<m>:CSSTate?....1862
SEARch:RESult:LIN:FRAMe<m>:CSValue?....1862
SEARch:RESult:LIN:FRAMe<m>:DATA?....1861
SEARch:RESult:LIN:FRAMe<m>:IDPValue?....1863
SEARch:RESult:LIN:FRAMe<m>:IDSTate?....1863
SEARch:RESult:LIN:FRAMe<m>:IDValue?....1863
SEARch:RESult:LIN:FRAMe<m>:SDATa?....1861
SEARch:RESult:LIN:FRAMe<m>:SDEXport?....1862
SEARch:RESult:LIN:FRAMe<m>:STARt?....1860
SEARch:RESult:LIN:FRAMe<m>:STATus?....1860
SEARch:RESult:LIN:FRAMe<m>:STOP?....1861
SEARch:RESult:LIN:FRAMe<m>:SYMBol?....1864
SEARch:RESult:LIN:FRAMe<m>:SYSTate?....1864
SEARch:RESult:LIN:FRAMe<m>:VERSion?....1864
SEARch:RESult:MDIO:WCOunt?....2055
SEARch:RESult:MDIO:WORD<m>:DATA?....2051
SEARch:RESult:MDIO:WORD<m>:PHYS?....2052
SEARch:RESult:MDIO:WORD<m>:REGI?....2052
SEARch:RESult:MDIO:WORD<m>:ST?....2052

SEARch:RESult:MDIO:WORD<m>:STARt?....2053
SEARch:RESult:MDIO:WORD<m>:STATe?....2053
SEARch:RESult:MDIO:WORD<m>:STOP?....2054
SEARch:RESult:MDIO:WORD<m>:SYMBol?....2054
SEARch:RESult:MDIO:WORD<m>:TYPE?....2054
SEARch:RESult:MILStd:WCOunt?....1980
SEARch:RESult:MILStd:WORD<m>:DATA?....1982
SEARch:RESult:MILStd:WORD<m>:INFO?....1982
SEARch:RESult:MILStd:WORD<m>:RTADdress?....1982
SEARch:RESult:MILStd:WORD<m>:STARt?....1981
SEARch:RESult:MILStd:WORD<m>:STATus?....1980
SEARch:RESult:MILStd:WORD<m>:STOP?....1981
SEARch:RESult:MILStd:WORD<m>:SYMBol?....1981
SEARch:RESult:MILStd:WORD<m>:TYPE?....1980
SEARch:RESult:MPHY:FCOunt?....2315
SEARch:RESult:MPHY:FRAMe<m>:CCOunt?....2315
SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:DATA?....2316
SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:STARt?....2316
SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:STATe?....2317
SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:STOP?....2317
SEARch:RESult:MPHY:FRAMe<m>:CELL<n>:TYPE?....2316
SEARch:RESult:MPHY:FRAMe<m>:DATA?....2317
SEARch:RESult:MPHY:FRAMe<m>:FTYPe?....2318
SEARch:RESult:MPHY:FRAMe<m>:STARt?....2319
SEARch:RESult:MPHY:FRAMe<m>:STATe?....2319
SEARch:RESult:MPHY:FRAMe<m>:STOP?....2320
SEARch:RESult:PCIE:PACKet<m>:ADDR?....2381
SEARch:RESult:PCIE:PACKet<m>:CPID?....2381
SEARch:RESult:PCIE:PACKet<m>:CPS?....2381
SEARch:RESult:PCIE:PACKet<m>:DATA?....2381
SEARch:RESult:PCIE:PACKet<m>:DFC?....2382
SEARch:RESult:PCIE:PACKet<m>:FCOunt?....2382
SEARch:RESult:PCIE:PACKet<m>:FLD<n>:FVAL?....2382
SEARch:RESult:PCIE:PACKet<m>:FLD<n>:STATus?....2382
SEARch:RESult:PCIE:PACKet<m>:FLD<n>:TYPE?....2383
SEARch:RESult:PCIE:PACKet<m>:FLD<n>:VAL?....2383
SEARch:RESult:PCIE:PACKet<m>:HFC?....2383
SEARch:RESult:PCIE:PACKet<m>:LEN?....2384
SEARch:RESult:PCIE:PACKet<m>:RQID?....2384
SEARch:RESult:PCIE:PACKet<m>:SEQN?....2384
SEARch:RESult:PCIE:PACKet<m>:STARt?....2385
SEARch:RESult:PCIE:PACKet<m>:STATe?....2385
SEARch:RESult:PCIE:PACKet<m>:STOP?....2385
SEARch:RESult:PCIE:PACKet<m>:TAG?....2385
SEARch:RESult:PCIE:PACKet<m>:TYPE?....2386
SEARch:RESult:PCIE:PCOunt?....2386
SEARch:RESult:RFFE:FCOunt?....2083
SEARch:RESult:RFFE:SEQuence<m>:ADDRess?....2084
SEARch:RESult:RFFE:SEQuence<m>:BCOunt?....2085
SEARch:RESult:RFFE:SEQuence<m>:BYTE<n>:STATe?....2085

SEARch:RESult:RFFE:SEQuence<m>:BYTE<n>:VALue?.....2086
SEARch:RESult:RFFE:SEQuence<m>:DATA?.....2085
SEARch:RESult:RFFE:SEQuence<m>:SADD?.....2084
SEARch:RESult:RFFE:SEQuence<m>:STARt?.....2083
SEARch:RESult:RFFE:SEQuence<m>:STATe?.....2083
SEARch:RESult:RFFE:SEQuence<m>:STOP?.....2084
SEARch:RESult:RFFE:SEQuence<m>:SYMBol?.....2085
SEARch:RESult:RFFE:SEQuence<m>:TYPE?.....2083
SEARch:RESult:SENT:FCOunt?.....1925
SEARch:RESult:SENT:FRAMe<m>:CSValue?.....1925
SEARch:RESult:SENT:FRAMe<m>:DATA?.....1925
SEARch:RESult:SENT:FRAMe<m>:IDTYpe?.....1925
SEARch:RESult:SENT:FRAMe<m>:IDValue?.....1926
SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:STATe?.....1926
SEARch:RESult:SENT:FRAMe<m>:NIBBle<n>:VALue?.....1926
SEARch:RESult:SENT:FRAMe<m>:PAPTicks?.....1927
SEARch:RESult:SENT:FRAMe<m>:SCOM?.....1927
SEARch:RESult:SENT:FRAMe<m>:SDATa?.....1927
SEARch:RESult:SENT:FRAMe<m>:STARt?.....1928
SEARch:RESult:SENT:FRAMe<m>:STATus?.....1928
SEARch:RESult:SENT:FRAMe<m>:STOP?.....1928
SEARch:RESult:SENT:FRAMe<m>:SYMBol?.....1928
SEARch:RESult:SENT:FRAMe<m>:SYNCduration?.....1929
SEARch:RESult:SENT:FRAMe<m>:TYPE?.....1929
SEARch:RESult:SHOW.....1638
SEARch:RESult:SORT:ASCending.....1638
SEARch:RESult:SORT[:MODE].....1639
SEARch:RESult:SPI:FCOunt?.....1724
SEARch:RESult:SPI:FRAMe<m>:COUNt?.....1724
SEARch:RESult:SPI:FRAMe<m>:DATA?.....1724
SEARch:RESult:SPI:FRAMe<m>:STARt?.....1724
SEARch:RESult:SPI:FRAMe<m>:STATus?.....1725
SEARch:RESult:SPI:FRAMe<m>:STOP?.....1725
SEARch:RESult:SPI:FRAMe<m>:WCOunt?.....1725
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:MISO?.....1726
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:MOSI?.....1726
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:STARt?.....1726
SEARch:RESult:SPI:FRAMe<m>:WORD<n>:STOP?.....1727
SEARch:RESult:SWIRe:FCOunt?.....2014
SEARch:RESult:SWIRe:FRAMe<m>:DATA?.....2014
SEARch:RESult:SWIRe:FRAMe<m>:STARt?.....2015
SEARch:RESult:SWIRe:FRAMe<m>:STATe?.....2015
SEARch:RESult:SWIRe:FRAMe<m>:STOP?.....2015
SEARch:RESult:SWIRe:FRAMe<m>:TYPE?.....2016
SEARch:RESult:TBTO:FCOunt?.....2142
SEARch:RESult:TBTO:FRAMe<m>:CRC?.....2142
SEARch:RESult:TBTO:FRAMe<m>:DATA?.....2142
SEARch:RESult:TBTO:FRAMe<m>:DESTaddress?.....2142
SEARch:RESult:TBTO:FRAMe<m>:DSYMbol?.....2143
SEARch:RESult:TBTO:FRAMe<m>:FTYPe?.....2143

SEARch:RESult:TBTO:FRAMe<m>:NUMWords?....2144
SEARch:RESult:TBTO:FRAMe<m>:SRCaddress?....2144
SEARch:RESult:TBTO:FRAMe<m>:SSYMbol?....2144
SEARch:RESult:TBTO:FRAMe<m>:STARt?....2145
SEARch:RESult:TBTO:FRAMe<m>:STATe?....2145
SEARch:RESult:TBTO:FRAMe<m>:STOP?....2145
SEARch:RESult:TBTO:FRAMe<m>:TYPE?....2146
SEARch:RESult:TBTO:FRAMe<m>:WORD<n>:TYPE?....2146
SEARch:RESult:TBTO:FRAMe<m>:WORD<n>:VALue?....2146
SEARch:RESult:TBTO:FRAMe<m>:WORD<n>:VSTR?....2147
SEARch:RESult:USB:PACKet<m>:ADDRess?....2183
SEARch:RESult:USB:PACKet<m>:CRC?....2183
SEARch:RESult:USB:PACKet<m>:DATA?....2184
SEARch:RESult:USB:PACKet<m>:ENDPoint?....2184
SEARch:RESult:USB:PACKet<m>:ET?....2184
SEARch:RESult:USB:PACKet<m>:FRAMe?....2184
SEARch:RESult:USB:PACKet<m>:PID?....2185
SEARch:RESult:USB:PACKet<m>:PORT?....2185
SEARch:RESult:USB:PACKet<m>:SC?....2186
SEARch:RESult:USB:PACKet<m>:SEU?....2186
SEARch:RESult:USB:PACKet<m>:STARt?....2186
SEARch:RESult:USB:PACKet<m>:STATus?....2187
SEARch:RESult:USB:PACKet<m>:STOP?....2187
SEARch:RESult:USB:PCOunt?....2187
SEARch:RESult:USBPd:FCOunt?....2238
SEARch:RESult:USBPd:FRAMe<m>:DATA?....2238
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:FVAL?....2238
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STARt?....2239
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STATus?....2239
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:STOP?....2239
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:TYPE?....2240
SEARch:RESult:USBPd:FRAMe<m>:FLD<n>:VAL?....2240
SEARch:RESult:USBPd:FRAMe<m>:INFO?....2241
SEARch:RESult:USBPd:FRAMe<m>:STARt?....2241
SEARch:RESult:USBPd:FRAMe<m>:STATe?....2241
SEARch:RESult:USBPd:FRAMe<m>:STOP?....2242
SEARch:RESult:USBPd:FRAMe<m>:TYPE?....2242
SEARch:RESult:USBThree:FCOunt?....2215
SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:FVAL?....2218
SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:STARt?....2217
SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:STATus?....2216
SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:STOP?....2217
SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:TYPE?....2216
SEARch:RESult:USBThree:FRAMe<m>:FLD<n>:VAL?....2218
SEARch:RESult:USBThree:FRAMe<m>:INFO?....2214
SEARch:RESult:USBThree:FRAMe<m>:STARt?....2215
SEARch:RESult:USBThree:FRAMe<m>:STATe?....2215
SEARch:RESult:USBThree:FRAMe<m>:STOP?....2216
SEARch:RESult:USBThree:FRAMe<m>:TYPE?....2214
SEARch:RESult[:ALL]?....1639

SEARch:SOURce.....1605
SEARch:TRIGger:ARINc:DATA:CONDition.....1994
SEARch:TRIGger:ARINc:DATA:MAX.....1994
SEARch:TRIGger:ARINc:DATA:MIN.....1994
SEARch:TRIGger:ARINc:ERRor:CODing.....1995
SEARch:TRIGger:ARINc:ERRor:PARity.....1995
SEARch:TRIGger:ARINc:ERRor:TIMing.....1996
SEARch:TRIGger:ARINc:LABel:CONDition.....1994
SEARch:TRIGger:ARINc:LABel:MAX.....1994
SEARch:TRIGger:ARINc:LABel:MIN.....1994
SEARch:TRIGger:ARINc:SDI.....1995
SEARch:TRIGger:ARINc:SSM.....1995
SEARch:TRIGger:ARINc:TYPE.....1993
SEARch:TRIGger:CAN:ACKerror.....1823
SEARch:TRIGger:CAN:BITSterror.....1824
SEARch:TRIGger:CAN:CRCerror.....1824
SEARch:TRIGger:CAN:DCONdition.....1820
SEARch:TRIGger:CAN:DLC.....1822
SEARch:TRIGger:CAN:DLCCondition.....1821
SEARch:TRIGger:CAN:DMAX.....1821
SEARch:TRIGger:CAN:DMIN.....1821
SEARch:TRIGger:CAN:FDATa:BRS.....1825
SEARch:TRIGger:CAN:FDATa:DPOPerator.....1822
SEARch:TRIGger:CAN:FDATa:DPOSition.....1823
SEARch:TRIGger:CAN:FDATa:DPTO.....1823
SEARch:TRIGger:CAN:FDATa:ESI.....1825
SEARch:TRIGger:CAN:FDATa:SCERror.....1824
SEARch:TRIGger:CAN:FDATa:STANdard.....1819
SEARch:TRIGger:CAN:FDATa[:FDF].....1825
SEARch:TRIGger:CAN:FORMerror.....1824
SEARch:TRIGger:CAN:FTYPe.....1819
SEARch:TRIGger:CAN:ICONdition.....1819
SEARch:TRIGger:CAN:IMAX.....1820
SEARch:TRIGger:CAN:IMIN.....1820
SEARch:TRIGger:CAN:ITYPe.....1819
SEARch:TRIGger:CAN:SERRor.....1818
SEARch:TRIGger:CAN:SFIDentifier.....1818
SEARch:TRIGger:CAN:SFTYpe.....1818
SEARch:TRIGger:CAN:SIDData.....1818
SEARch:TRIGger:CAN:SSYMbolic.....1834
SEARch:TRIGger:CAN:SYMBolic:DMAX.....1836
SEARch:TRIGger:CAN:SYMBolic:DMIN.....1836
SEARch:TRIGger:CAN:SYMBolic:MSGValue.....1835
SEARch:TRIGger:CAN:SYMBolic:SGEValue.....1836
SEARch:TRIGger:CAN:SYMBolic:SIGValue.....1835
SEARch:TRIGger:CAN:SYMBolic:SSIGnals.....1835
SEARch:TRIGger:CAN[:SSOFrame].....1817
SEARch:TRIGger:CMSB:BIT.....1785
SEARch:TRIGger:CMSB:DMAX.....1786
SEARch:TRIGger:CMSB:DMIN.....1786

SEARch:TRIGger:CMSB:DOPerator....1787
SEARch:TRIGger:CMSB:ERENable....1785
SEARch:TRIGger:CMSB:ERRor<m>:ENABle....1785
SEARch:TRIGger:CMSB:FIENable....1785
SEARch:TRIGger:CMSB:FRAMe<m>:ENABle....1787
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:BIT....1785
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMAX....1786
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DMIN....1786
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:DOPerator....1787
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:ENABle....1787
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMAX....1788
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IMIN....1788
SEARch:TRIGger:CMSB:FRAMe<m>:FLD<n>:IOPerator....1788
SEARch:TRIGger:CMSB:FRENable....1785
SEARch:TRIGger:CMSB:IMAX....1788
SEARch:TRIGger:CMSB:IMIN....1788
SEARch:TRIGger:CMSB:IOPerator....1788
SEARch:TRIGger:CXPI:CT....1945
SEARch:TRIGger:CXPI:DATA:DCONdition....1946
SEARch:TRIGger:CXPI:DATA:DMAX....1946
SEARch:TRIGger:CXPI:DATA:DMIN....1946
SEARch:TRIGger:CXPI:DATA:ICONdition....1947
SEARch:TRIGger:CXPI:DATA:IMAX....1947
SEARch:TRIGger:CXPI:DATA:IMIN....1947
SEARch:TRIGger:CXPI:DEXTension:CONDition....1948
SEARch:TRIGger:CXPI:DEXTension:MAX....1948
SEARch:TRIGger:CXPI:DEXTension:MIN....1948
SEARch:TRIGger:CXPI:DLC:CONDition....1949
SEARch:TRIGger:CXPI:DLC:MAX....1949
SEARch:TRIGger:CXPI:DLC:MIN....1949
SEARch:TRIGger:CXPI:ERRor:CRC....1949
SEARch:TRIGger:CXPI:ERRor:DLC....1950
SEARch:TRIGger:CXPI:ERRor:IBS....1950
SEARch:TRIGger:CXPI:ERRor:IFS....1950
SEARch:TRIGger:CXPI:ERRor:LENGth....1950
SEARch:TRIGger:CXPI:ERRor:PARity....1951
SEARch:TRIGger:CXPI:ERRor:UART....1951
SEARch:TRIGger:CXPI:FID:CONDition....1951
SEARch:TRIGger:CXPI:FID:MAX....1952
SEARch:TRIGger:CXPI:FID:MIN....1952
SEARch:TRIGger:CXPI:NM....1952
SEARch:TRIGger:CXPI:TYPE....1952
SEARch:TRIGger:DDR:DMAX....2395
SEARch:TRIGger:DDR:DMIN....2395
SEARch:TRIGger:DDR:DOPerator....2396
SEARch:TRIGger:DDR:ERENable....2394
SEARch:TRIGger:DDR:ERRor<m>:ENABle....2394
SEARch:TRIGger:DDR:FIENable....2394
SEARch:TRIGger:DDR:FRAMe<m>:ENABle....2396
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DMAX....2395

SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DMIN.....2395
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:DOPerator.....2396
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IMAX.....2397
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IMIN.....2397
SEARch:TRIGger:DDR:FRAMe<m>:FLD<n>:IOPerator.....2397
SEARch:TRIGger:DDR:FRENable.....2395
SEARch:TRIGger:DDR:FRENable.....2396
SEARch:TRIGger:DDR:IMAX.....2397
SEARch:TRIGger:DDR:IMIN.....2397
SEARch:TRIGger:DDR:IOPerator.....2397
SEARch:TRIGger:DPHY:DATA:CONDition.....2262
SEARch:TRIGger:DPHY:DATA:MAX.....2262
SEARch:TRIGger:DPHY:DATA:MIN.....2262
SEARch:TRIGger:DPHY:DIDX:CONDition.....2263
SEARch:TRIGger:DPHY:DIDX:MAX.....2263
SEARch:TRIGger:DPHY:DIDX:MIN.....2263
SEARch:TRIGger:DPHY:DTYPe:CONDition.....2264
SEARch:TRIGger:DPHY:DTYPe:MAX.....2264
SEARch:TRIGger:DPHY:DTYPe:MIN.....2264
SEARch:TRIGger:DPHY:ESCMode:CONDition.....2265
SEARch:TRIGger:DPHY:ESCMode:MAX.....2265
SEARch:TRIGger:DPHY:ESCMode:MIN.....2265
SEARch:TRIGger:DPHY:ESDTa:CONDition.....2266
SEARch:TRIGger:DPHY:ESDTa:MAX.....2266
SEARch:TRIGger:DPHY:ESDTa:MIN.....2266
SEARch:TRIGger:DPHY:ESINdex:CONDition.....2267
SEARch:TRIGger:DPHY:ESINdex:MAX.....2267
SEARch:TRIGger:DPHY:ESINdex:MIN.....2267
SEARch:TRIGger:DPHY:HSVC.....2268
SEARch:TRIGger:DPHY:TYPE.....2261
SEARch:TRIGger:DPHY:WCOunt:CONDition.....2268
SEARch:TRIGger:DPHY:WCOunt:MAX.....2268
SEARch:TRIGger:DPHY:WCOunt:MIN.....2269
SEARch:TRIGger:EBTB:DISParityerr.....2341
SEARch:TRIGger:EBTB:DX.....2342
SEARch:TRIGger:EBTB:DY.....2342
SEARch:TRIGger:EBTB:EBPA.....2342
SEARch:TRIGger:EBTB:GLITcherror.....2342
SEARch:TRIGger:EBTB:STYPe.....2343
SEARch:TRIGger:EBTB:SYME.....2343
SEARch:TRIGger:EBTB:SYMType.....2343
SEARch:TRIGger:EBTB:TBPA.....2344
SEARch:TRIGger:EBTB:TYPE.....2344
SEARch:TRIGger:EBTB:UNK.....2344
SEARch:TRIGger:EDGE:ACOPy.....1607
SEARch:TRIGger:EDGE:BCOPy.....1608
SEARch:TRIGger:EDGE:SLOPe.....1608
SEARch:TRIGger:EDGE[:STATe].....1606
SEARch:TRIGger:ETHernet:ERRor:LENGth.....2035
SEARch:TRIGger:ETHernet:ERRor:PREamble.....2035

SEARch:TRIGger:ETHernet:FRAMe:CCONdition....2034
SEARch:TRIGger:ETHernet:FRAMe:CMAX....2034
SEARch:TRIGger:ETHernet:FRAMe:CMIN....2034
SEARch:TRIGger:ETHernet:FRAMe:DCONdition....2031
SEARch:TRIGger:ETHernet:FRAMe:DMAX....2031
SEARch:TRIGger:ETHernet:FRAMe:DMIN....2031
SEARch:TRIGger:ETHernet:FRAMe:SCONdition....2032
SEARch:TRIGger:ETHernet:FRAMe:SMAX....2032
SEARch:TRIGger:ETHernet:FRAMe:SMIN....2032
SEARch:TRIGger:ETHernet:FRAMe:TCONdition....2033
SEARch:TRIGger:ETHernet:FRAMe:TMAX....2033
SEARch:TRIGger:ETHernet:FRAMe:TMIN....2033
SEARch:TRIGger:ETHernet:TYPE....2030
SEARch:TRIGger:FLXRay:BSSerror....1892
SEARch:TRIGger:FLXRay:CENable....1885
SEARch:TRIGger:FLXRay:CMAX....1886
SEARch:TRIGger:FLXRay:CMIN....1886
SEARch:TRIGger:FLXRay:CSTep....1886
SEARch:TRIGger:FLXRay:DCONdition....1887
SEARch:TRIGger:FLXRay:DMAX....1887
SEARch:TRIGger:FLXRay:DMIN....1887
SEARch:TRIGger:FLXRay:DPOPerator....1888
SEARch:TRIGger:FLXRay:DPOSition....1888
SEARch:TRIGger:FLXRay:DPTO....1888
SEARch:TRIGger:FLXRay:FCONdition....1889
SEARch:TRIGger:FLXRay:FESerror....1892
SEARch:TRIGger:FLXRay:FMAX....1889
SEARch:TRIGger:FLXRay:FMIN....1889
SEARch:TRIGger:FLXRay:FSSerror....1893
SEARch:TRIGger:FLXRay:HCRCerror....1893
SEARch:TRIGger:FLXRay:NUFRame....1890
SEARch:TRIGger:FLXRay:PCONdition....1890
SEARch:TRIGger:FLXRay:PCRCerror....1893
SEARch:TRIGger:FLXRay:PLPReamble....1890
SEARch:TRIGger:FLXRay:PMAX....1891
SEARch:TRIGger:FLXRay:PMIN....1891
SEARch:TRIGger:FLXRay:SERRor....1884
SEARch:TRIGger:FLXRay:SIDData....1885
SEARch:TRIGger:FLXRay:SSYMbol....1885
SEARch:TRIGger:FLXRay:STFRame....1891
SEARch:TRIGger:FLXRay:SYFRame....1892
SEARch:TRIGger:FLXRay:SYMBol....1892
SEARch:TRIGger:FLXRay[:SSOFrame]....1885
SEARch:TRIGger:GLITch:ACOPy....1607
SEARch:TRIGger:GLITch:POLarity....1609
SEARch:TRIGger:GLITch:RANGe....1609
SEARch:TRIGger:GLITch:WIDTh....1609
SEARch:TRIGger:GLITch[:STATe]....1606
SEARch:TRIGger:HBTO:CRC:CONDition....2107
SEARch:TRIGger:HBTO:CRC:MAX....2108

SEARch:TRIGger:HBTO:CRC:MIN 2107
SEARch:TRIGger:HBTO:DADDress:CONDition 2104
SEARch:TRIGger:HBTO:DADDress:MAX 2105
SEARch:TRIGger:HBTO:DADDress:MIN 2104
SEARch:TRIGger:HBTO:DATA:DCONdition 2108
SEARch:TRIGger:HBTO:DATA:DMAX 2109
SEARch:TRIGger:HBTO:DATA:DMIN 2108
SEARch:TRIGger:HBTO:DATA:ICONdition 2109
SEARch:TRIGger:HBTO:DATA:IMAX 2110
SEARch:TRIGger:HBTO:DATA:IMIN 2109
SEARch:TRIGger:HBTO:ERRor:CRC 2110
SEARch:TRIGger:HBTO:ERRor:PREamble 2110
SEARch:TRIGger:HBTO:ERRor:SFD 2110
SEARch:TRIGger:HBTO:LENGth:CONDition 2106
SEARch:TRIGger:HBTO:LENGth:MAX 2107
SEARch:TRIGger:HBTO:LENGth:MIN 2106
SEARch:TRIGger:HBTO:SADDress:CONDition 2105
SEARch:TRIGger:HBTO:SADDress:MAX 2106
SEARch:TRIGger:HBTO:SADDress:MIN 2105
SEARch:TRIGger:HBTO:TYPE 2103
SEARch:TRIGger:I2C:ACCess 1697
SEARch:TRIGger:I2C:ACONdition 1695
SEARch:TRIGger:I2C:ADData 1695
SEARch:TRIGger:I2C:ADDO<m>:ADRType 1697
SEARch:TRIGger:I2C:ADDO<m>:ENABle 1697
SEARch:TRIGger:I2C:ADDO<m>:RWBit 1698
SEARch:TRIGger:I2C:ADDO<m>[:VALue] 1698
SEARch:TRIGger:I2C:ADDRess 1696
SEARch:TRIGger:I2C:ADDTo 1696
SEARch:TRIGger:I2C:ADNack 1700
SEARch:TRIGger:I2C:ADOR 1695
SEARch:TRIGger:I2C:AMODe 1696
SEARch:TRIGger:I2C:DCONdition 1699
SEARch:TRIGger:I2C:DMAX 1700
SEARch:TRIGger:I2C:DMIN 1700
SEARch:TRIGger:I2C:DPOPerator 1698
SEARch:TRIGger:I2C:DPOSition 1699
SEARch:TRIGger:I2C:DPTO 1699
SEARch:TRIGger:I2C:DRNack 1700
SEARch:TRIGger:I2C:DWNack 1701
SEARch:TRIGger:I2C:NACKnowledge 1694
SEARch:TRIGger:I2C:RCONdition 1693
SEARch:TRIGger:I2C:SADDress 1694
SEARch:TRIGger:I2C:SCONdition 1693
SEARch:TRIGger:I2C:STCNdition 1694
SEARch:TRIGger:INTerval:ACOPy 1607
SEARch:TRIGger:INTerval:DELTa 1610
SEARch:TRIGger:INTerval:RANGe 1610
SEARch:TRIGger:INTerval:SLOPe 1610
SEARch:TRIGger:INTerval:WIDTh 1611

SEARch:TRIGger:INTerval[:STATe]....1606
SEARch:TRIGger:LEVel:RUNT:LOWer....1613
SEARch:TRIGger:LEVel:RUNT:UPPer....1613
SEARch:TRIGger:LEVel:TRANsition:LOWer....1615
SEARch:TRIGger:LEVel:TRANsition:UPPer....1615
SEARch:TRIGger:LEVel:WINDow:LOWer....1621
SEARch:TRIGger:LEVel:WINDow:UPPer....1621
SEARch:TRIGger:LEVel[:VALue]....1607
SEARch:TRIGger:LIN:BORDer....1857
SEARch:TRIGger:LIN:CHKSerror....1858
SEARch:TRIGger:LIN:CRCDatalen....1859
SEARch:TRIGger:LIN:DCONdition....1856
SEARch:TRIGger:LIN:DLECondition....1857
SEARch:TRIGger:LIN:DLENgth....1858
SEARch:TRIGger:LIN:DMAX....1856
SEARch:TRIGger:LIN:DMIN....1856
SEARch:TRIGger:LIN:ERRPattern....1859
SEARch:TRIGger:LIN:ICONdition....1854
SEARch:TRIGger:LIN:IDENtifieror....1853
SEARch:TRIGger:LIN:IDOR<m>:ENABle....1855
SEARch:TRIGger:LIN:IDOR<m>[:VALue]....1855
SEARch:TRIGger:LIN:IMAX....1855
SEARch:TRIGger:LIN:IMIN....1855
SEARch:TRIGger:LIN:IPERror....1858
SEARch:TRIGger:LIN:SERRor....1854
SEARch:TRIGger:LIN:SFIDentifier....1853
SEARch:TRIGger:LIN:SIDData....1853
SEARch:TRIGger:LIN:SSOFrame....1853
SEARch:TRIGger:LIN:STANdard....1859
SEARch:TRIGger:LIN:SYERror....1858
SEARch:TRIGger:LIN:WUFRame....1854
SEARch:TRIGger:MDIO:DATA....2049
SEARch:TRIGger:MDIO:FRAMetype....2049
SEARch:TRIGger:MDIO:PHYS....2050
SEARch:TRIGger:MDIO:REGI....2050
SEARch:TRIGger:MDIO:ST....2050
SEARch:TRIGger:MDIO:TYPE....2051
SEARch:TRIGger:MILStd:CDST:ICONdition....1976
SEARch:TRIGger:MILStd:CDST:IMAX....1977
SEARch:TRIGger:MILStd:CDST:IMIN....1976
SEARch:TRIGger:MILStd:CDST:RCONdition....1974
SEARch:TRIGger:MILStd:CDST:RMAX....1975
SEARch:TRIGger:MILStd:CDST:RMIN....1975
SEARch:TRIGger:MILStd:CMD:CCONdition....1976
SEARch:TRIGger:MILStd:CMD:CMAX....1977
SEARch:TRIGger:MILStd:CMD:CMIN....1976
SEARch:TRIGger:MILStd:CMD:RCONdition....1974
SEARch:TRIGger:MILStd:CMD:RMAX....1975
SEARch:TRIGger:MILStd:CMD:RMIN....1975
SEARch:TRIGger:MILStd:CMD:SCONdition....1976

SEARch:TRIGger:MILStd:CMD:SMAX....1977
SEARch:TRIGger:MILStd:CMD:SMIN....1976
SEARch:TRIGger:MILStd:CMD:TR....1978
SEARch:TRIGger:MILStd:DATA:DCONdition....1976
SEARch:TRIGger:MILStd:DATA:DMAX....1977
SEARch:TRIGger:MILStd:DATA:DMIN....1976
SEARch:TRIGger:MILStd:DATA:ICONdition....1977
SEARch:TRIGger:MILStd:DATA:IMAX....1978
SEARch:TRIGger:MILStd:DATA:IMIN....1977
SEARch:TRIGger:MILStd:DATA:RCONdition....1974
SEARch:TRIGger:MILStd:DATA:RMAX....1975
SEARch:TRIGger:MILStd:DATA:RMIN....1975
SEARch:TRIGger:MILStd:ERRor:MANChester....1978
SEARch:TRIGger:MILStd:ERRor:PARity....1978
SEARch:TRIGger:MILStd:ERRor:SYNC....1978
SEARch:TRIGger:MILStd:ERRor:TIMing....1978
SEARch:TRIGger:MILStd:STATus:BCReceived....1979
SEARch:TRIGger:MILStd:STATus:BUSY....1979
SEARch:TRIGger:MILStd:STATus:DBCaccept....1979
SEARch:TRIGger:MILStd:STATus:INSTrument....1979
SEARch:TRIGger:MILStd:STATus:MERRor....1979
SEARch:TRIGger:MILStd:STATus:SREQuest....1979
SEARch:TRIGger:MILStd:STATus:SUBSystem....1979
SEARch:TRIGger:MILStd:STATus:TERMinal....1979
SEARch:TRIGger:MILStd:TPSPecifier....1979
SEARch:TRIGger:MILStd:TYPE....1974
SEARch:TRIGger:MPHY:CRC:CONDition....2309
SEARch:TRIGger:MPHY:CRC:MAX....2310
SEARch:TRIGger:MPHY:CRC:MIN....2309
SEARch:TRIGger:MPHY:CREDit:CONDition....2311
SEARch:TRIGger:MPHY:CREDit:MAX....2311
SEARch:TRIGger:MPHY:CREDit:MIN....2311
SEARch:TRIGger:MPHY:CREQ....2310
SEARch:TRIGger:MPHY:DATA:DCONdition....2306
SEARch:TRIGger:MPHY:DATA:DMAX....2307
SEARch:TRIGger:MPHY:DATA:DMIN....2306
SEARch:TRIGger:MPHY:DATA:ICONdition....2307
SEARch:TRIGger:MPHY:DATA:IMAX....2308
SEARch:TRIGger:MPHY:DATA:IMIN....2307
SEARch:TRIGger:MPHY:ERRor:LCMD....2314
SEARch:TRIGger:MPHY:ERRor:LENGth....2315
SEARch:TRIGger:MPHY:ERRor:REServed....2314
SEARch:TRIGger:MPHY:ERRor:SYMBol....2314
SEARch:TRIGger:MPHY:ERRor:UNKNown....2314
SEARch:TRIGger:MPHY:FSNumber:CONDition....2308
SEARch:TRIGger:MPHY:FSNumber:MAX....2309
SEARch:TRIGger:MPHY:FSNumber:MIN....2308
SEARch:TRIGger:MPHY:LCCType....2301
SEARch:TRIGger:MPHY:LWFour:CONDition....2305
SEARch:TRIGger:MPHY:LWFour:MAX....2305

SEARch:TRIGger:MPHY:LWFour:MIN.....2305
SEARch:TRIGger:MPHY:LWONe:CONDition.....2302
SEARch:TRIGger:MPHY:LWONe:MAX.....2302
SEARch:TRIGger:MPHY:LWONe:MIN.....2302
SEARch:TRIGger:MPHY:LWTHree:CONDition.....2304
SEARch:TRIGger:MPHY:LWTHree:MAX.....2304
SEARch:TRIGger:MPHY:LWTHree:MIN.....2304
SEARch:TRIGger:MPHY:LWTWo:CONDition.....2303
SEARch:TRIGger:MPHY:LWTWo:MAX.....2303
SEARch:TRIGger:MPHY:LWTWo:MIN.....2303
SEARch:TRIGger:MPHY:PBEGin:CONDition.....2312
SEARch:TRIGger:MPHY:PBEGin:MAX.....2312
SEARch:TRIGger:MPHY:PBEGin:MIN.....2312
SEARch:TRIGger:MPHY:PFID:CONDition.....2313
SEARch:TRIGger:MPHY:PFID:MAX.....2313
SEARch:TRIGger:MPHY:PFID:MIN.....2313
SEARch:TRIGger:MPHY:RREQ.....2310
SEARch:TRIGger:MPHY:SSIC:BIT.....2327
SEARch:TRIGger:MPHY:SSIC:DMAX.....2328
SEARch:TRIGger:MPHY:SSIC:DMIN.....2328
SEARch:TRIGger:MPHY:SSIC:DOPerator.....2329
SEARch:TRIGger:MPHY:SSIC:ERENable.....2325
SEARch:TRIGger:MPHY:SSIC:ERRor<m>:ENABle.....2325
SEARch:TRIGger:MPHY:SSIC:FIENable.....2326
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:ENABle.....2326
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:BIT.....2327
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DMAX.....2328
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DMIN.....2328
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:DOPerator.....2329
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:ENABle.....2326
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:ENABle.....2327
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IMAX.....2330
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IMIN.....2330
SEARch:TRIGger:MPHY:SSIC:FRAMe<m>:FLD<n>:IOPerator.....2330
SEARch:TRIGger:MPHY:SSIC:FRENable.....2326
SEARch:TRIGger:MPHY:SSIC:FRENable.....2329
SEARch:TRIGger:MPHY:SSIC:IMAX.....2330
SEARch:TRIGger:MPHY:SSIC:IMIN.....2330
SEARch:TRIGger:MPHY:SSIC:IOPerator.....2330
SEARch:TRIGger:MPHY:TC.....2306
SEARch:TRIGger:MPHY:TYPE.....2300
SEARch:TRIGger:PATTern:A:LOGic.....1624
SEARch:TRIGger:PATTern:A[:ENABle].....1624
SEARch:TRIGger:PATTern:AB:LOGic.....1625
SEARch:TRIGger:PATTern:ABCD:LOGic.....1625
SEARch:TRIGger:PATTern:ACOPy.....1607
SEARch:TRIGger:PATTern:B:LOGic.....1624
SEARch:TRIGger:PATTern:B[:ENABle].....1624
SEARch:TRIGger:PATTern:C:LOGic.....1624
SEARch:TRIGger:PATTern:C[:ENABle].....1624

SEARch:TRIGger:PATTern:CD:LOGic....1625
SEARch:TRIGger:PATTern:D:LOGic....1624
SEARch:TRIGger:PATTern:D[:ENABle]....1624
SEARch:TRIGger:PATTern:MODE....1625
SEARch:TRIGger:PATTern:TIMeout:MODE....1626
SEARch:TRIGger:PATTern:TIMeout[:TIME]....1626
SEARch:TRIGger:PATTern:WIDTh:DELTa....1627
SEARch:TRIGger:PATTern:WIDTh:RANGe....1626
SEARch:TRIGger:PATTern:WIDTh[:WIDTh]....1627
SEARch:TRIGger:PATTern[:STATe]....1606
SEARch:TRIGger:PCIE:DLLP:FCTL:CRET....2369
SEARch:TRIGger:PCIE:DLLP:FCTL:VCID....2369
SEARch:TRIGger:PCIE:DLLP:MULT:AUTH....2369
SEARch:TRIGger:PCIE:DLLP:MULT:CRET....2369
SEARch:TRIGger:PCIE:DLLP:MULT:DVPT....2370
SEARch:TRIGger:PCIE:DLLP:MULT:HABS....2370
SEARch:TRIGger:PCIE:DLLP:MULT:MIXT....2370
SEARch:TRIGger:PCIE:DLLP:MULT:PHAS....2370
SEARch:TRIGger:PCIE:DLLP:MULT:RESA....2371
SEARch:TRIGger:PCIE:DLLP:MULT:TLPT....2371
SEARch:TRIGger:PCIE:DLLP:MULT:TYPE....2371
SEARch:TRIGger:PCIE:DLLP:MULT:VHFC....2371
SEARch:TRIGger:PCIE:DLLP:MULT:VHGR....2372
SEARch:TRIGger:PCIE:DLLP:MULT:VLNR....2372
SEARch:TRIGger:PCIE:DLLP:POWM....2372
SEARch:TRIGger:PCIE:DLLP:SEQ....2373
SEARch:TRIGger:PCIE:DLLP:TYPE....2373
SEARch:TRIGger:PCIE:DLLP:VPAT....2373
SEARch:TRIGger:PCIE:ERRC:CRC....2373
SEARch:TRIGger:PCIE:ERRC:DISP....2374
SEARch:TRIGger:PCIE:ERRC:ECRC....2374
SEARch:TRIGger:PCIE:ERRC:INVP....2374
SEARch:TRIGger:PCIE:ERRC:LCRC....2374
SEARch:TRIGger:PCIE:OSET:COMP....2375
SEARch:TRIGger:PCIE:OSET:EIDE....2375
SEARch:TRIGger:PCIE:OSET:EIDL....2375
SEARch:TRIGger:PCIE:OSET:FTS....2375
SEARch:TRIGger:PCIE:OSET:SKIP....2376
SEARch:TRIGger:PCIE:OSET:TSONe....2376
SEARch:TRIGger:PCIE:OSET:TSTWo....2376
SEARch:TRIGger:PCIE:TLP:ADRT....2376
SEARch:TRIGger:PCIE:TLP:CFGT....2377
SEARch:TRIGger:PCIE:TLP:CPID....2377
SEARch:TRIGger:PCIE:TLP:CPLS....2377
SEARch:TRIGger:PCIE:TLP:DEID....2377
SEARch:TRIGger:PCIE:TLP:MERW....2378
SEARch:TRIGger:PCIE:TLP:MSGC....2378
SEARch:TRIGger:PCIE:TLP:MSGR....2378
SEARch:TRIGger:PCIE:TLP:ORDE....2378
SEARch:TRIGger:PCIE:TLP:REID....2379

SEARch:TRIGger:PCIE:TLP:SNOO....2379
SEARch:TRIGger:PCIE:TLP:SNUM....2379
SEARch:TRIGger:PCIE:TLP:TCHN....2379
SEARch:TRIGger:PCIE:TLP:TYPE....2380
SEARch:TRIGger:PCIE:TYPE....2380
SEARch:TRIGger:RFFE:ADDRess:CONDition....2078
SEARch:TRIGger:RFFE:ADDRess:MAX....2080
SEARch:TRIGger:RFFE:ADDRess:MIN....2080
SEARch:TRIGger:RFFE:BCOunt:CONDition....2079
SEARch:TRIGger:RFFE:BCOunt:MAX....2080
SEARch:TRIGger:RFFE:BCOunt:MIN....2080
SEARch:TRIGger:RFFE:DATA:DCON....2079
SEARch:TRIGger:RFFE:DATA:DMAX....2080
SEARch:TRIGger:RFFE:DATA:DMIN....2080
SEARch:TRIGger:RFFE:DATA:ICONdition....2079
SEARch:TRIGger:RFFE:DATA:IMAX....2080
SEARch:TRIGger:RFFE:DATA:IMIN....2080
SEARch:TRIGger:RFFE:ERRor:GAP....2082
SEARch:TRIGger:RFFE:ERRor:LENGth....2081
SEARch:TRIGger:RFFE:ERRor:PARity....2081
SEARch:TRIGger:RFFE:ERRor:SSC....2081
SEARch:TRIGger:RFFE:ERRor:VERSion....2082
SEARch:TRIGger:RFFE:INTerrupt....2080
SEARch:TRIGger:RFFE:SADD:CONDition....2078
SEARch:TRIGger:RFFE:SADD:MAX....2080
SEARch:TRIGger:RFFE:SADD:MIN....2080
SEARch:TRIGger:RFFE:TYPE....2076
SEARch:TRIGger:RUNT:ACOPy....1607
SEARch:TRIGger:RUNT:DELTa....1611
SEARch:TRIGger:RUNT:POLarity....1612
SEARch:TRIGger:RUNT:RANGe....1612
SEARch:TRIGger:RUNT:WIDTh....1613
SEARch:TRIGger:RUNT[:STATe]....1606
SEARch:TRIGger:SENT:CALibration....1917
SEARch:TRIGger:SENT:CRCerror....1924
SEARch:TRIGger:SENT:ERRor....1918
SEARch:TRIGger:SENT:FORMerror....1923
SEARch:TRIGger:SENT:IRFLength....1923
SEARch:TRIGger:SENT:PPERioderror....1923
SEARch:TRIGger:SENT:PULSeerror....1923
SEARch:TRIGger:SENT:SDCN....1922
SEARch:TRIGger:SENT:SDMN....1922
SEARch:TRIGger:SENT:SDMX....1922
SEARch:TRIGger:SENT:SICN....1921
SEARch:TRIGger:SENT:SIDType....1920
SEARch:TRIGger:SENT:SIMN....1921
SEARch:TRIGger:SENT:SIMX....1921
SEARch:TRIGger:SENT:SMSG....1917
SEARch:TRIGger:SENT:STATus....1919
SEARch:TRIGger:SENT:STYPe....1920

SEARch:TRIGger:SENT:TDCN.... 1919
SEARch:TRIGger:SENT:TDMN.... 1919
SEARch:TRIGger:SENT:TDMX.... 1920
SEARch:TRIGger:SENT:TRANsmission.... 1917
SEARch:TRIGger:SENT:TTYPe.... 1918
SEARch:TRIGger:SETHold:ACOPy.... 1607
SEARch:TRIGger:SETHold:CEDGe.... 1621
SEARch:TRIGger:SETHold:CLEVel.... 1622
SEARch:TRIGger:SETHold:CSOurce.... 1622
SEARch:TRIGger:SETHold:HTIMe.... 1622
SEARch:TRIGger:SETHold:STIMe.... 1623
SEARch:TRIGger:SETHold[:STATe].... 1606
SEARch:TRIGger:SLEWrate:ACOPy.... 1607
SEARch:TRIGger:SLEWrate:DELTa.... 1614
SEARch:TRIGger:SLEWrate:RANGe.... 1614
SEARch:TRIGger:SLEWrate:SLOPe.... 1615
SEARch:TRIGger:SLEWrate:TIME.... 1615
SEARch:TRIGger:SLEWrate[:STATe].... 1607
SEARch:TRIGger:SPI:DPOPerator.... 1722
SEARch:TRIGger:SPI:DPOSition.... 1722
SEARch:TRIGger:SPI:DPTO.... 1723
SEARch:TRIGger:SPI:FCONdition.... 1721
SEARch:TRIGger:SPI:MISopattern.... 1722
SEARch:TRIGger:SPI:MOSipattern.... 1722
SEARch:TRIGger:SPI:PALignment.... 1723
SEARch:TRIGger:SPI:TYPE.... 1721
SEARch:TRIGger:STATe:A:LOGic.... 1629
SEARch:TRIGger:STATe:A[:ENABle].... 1629
SEARch:TRIGger:STATe:AB:LOGic.... 1630
SEARch:TRIGger:STATe:ABCD:LOGic.... 1630
SEARch:TRIGger:STATe:ACOPy.... 1608
SEARch:TRIGger:STATe:B:LOGic.... 1629
SEARch:TRIGger:STATe:B[:ENABle].... 1629
SEARch:TRIGger:STATe:C:LOGic.... 1629
SEARch:TRIGger:STATe:C[:ENABle].... 1629
SEARch:TRIGger:STATe:CD:LOGic.... 1630
SEARch:TRIGger:STATe:CEDGe.... 1628
SEARch:TRIGger:STATe:CLEVel.... 1628
SEARch:TRIGger:STATe:CSOurce.... 1628
SEARch:TRIGger:STATe:D:LOGic.... 1629
SEARch:TRIGger:STATe:D[:ENABle].... 1629
SEARch:TRIGger:STATe[:STATe].... 1607
SEARch:TRIGger:SWIRe:CTYPe.... 2011
SEARch:TRIGger:SWIRe:DATA:CONDition.... 2011
SEARch:TRIGger:SWIRe:DATA:MAX.... 2012
SEARch:TRIGger:SWIRe:DATA:MIN.... 2012
SEARch:TRIGger:SWIRe:ERRor:ESC.... 2012
SEARch:TRIGger:SWIRe:ERRor:PARity.... 2012
SEARch:TRIGger:SWIRe:TIME:CONDition.... 2013
SEARch:TRIGger:SWIRe:TIME:MAX.... 2013

SEARch:TRIGger:SWIRe:TIME:MIN....2013
SEARch:TRIGger:SWIRe:TYPE....2014
SEARch:TRIGger:TBTO:CRC:CONDition....2134
SEARch:TRIGger:TBTO:CRC:MAX....2134
SEARch:TRIGger:TBTO:CRC:MIN....2135
SEARch:TRIGger:TBTO:DADDress:CONDition....2135
SEARch:TRIGger:TBTO:DADDress:MAX....2135
SEARch:TRIGger:TBTO:DADDress:MIN....2136
SEARch:TRIGger:TBTO:DATA:DCONdition....2136
SEARch:TRIGger:TBTO:DATA:DMAX....2136
SEARch:TRIGger:TBTO:DATA:DMIN....2137
SEARch:TRIGger:TBTO:DATA:ICONdition....2137
SEARch:TRIGger:TBTO:DATA:IMAX....2137
SEARch:TRIGger:TBTO:DATA:IMIN....2138
SEARch:TRIGger:TBTO:ERRor:CRC....2138
SEARch:TRIGger:TBTO:ERRor:FEC....2138
SEARch:TRIGger:TBTO:ERRor:OOR....2138
SEARch:TRIGger:TBTO:ERRor:ZERO....2139
SEARch:TRIGger:TBTO:LENGth:CONDition....2139
SEARch:TRIGger:TBTO:LENGth:MAX....2139
SEARch:TRIGger:TBTO:LENGth:MIN....2140
SEARch:TRIGger:TBTO:SADDress:CONDition....2140
SEARch:TRIGger:TBTO:SADDress:MAX....2140
SEARch:TRIGger:TBTO:SADDress:MIN....2141
SEARch:TRIGger:TBTO:TYPE....2141
SEARch:TRIGger:TIMeout:ACOPy....1608
SEARch:TRIGger:TIMeout:RANGe....1616
SEARch:TRIGger:TIMeout:TIME....1616
SEARch:TRIGger:TIMeout[:STATe]....1607
SEARch:TRIGger:USB:ACONdition....2169
SEARch:TRIGger:USB:AMAX....2170
SEARch:TRIGger:USB:AMIN....2169
SEARch:TRIGger:USB:BITSterror....2170
SEARch:TRIGger:USB:CRC5error....2170
SEARch:TRIGger:USB:CRC16error....2170
SEARch:TRIGger:USB:DATA....2171
SEARch:TRIGger:USB:DCONdition....2171
SEARch:TRIGger:USB:DPOPerator....2171
SEARch:TRIGger:USB:DPOSition....2172
SEARch:TRIGger:USB:ECONdition....2172
SEARch:TRIGger:USB:EMAX....2173
SEARch:TRIGger:USB:EMIN....2173
SEARch:TRIGger:USB:FCONdition....2173
SEARch:TRIGger:USB:FMAX....2174
SEARch:TRIGger:USB:FMIN....2173
SEARch:TRIGger:USB:GLITcherror....2174
SEARch:TRIGger:USB:HAND....2174
SEARch:TRIGger:USB:PATT....2174
SEARch:TRIGger:USB:PCONdition....2175
SEARch:TRIGger:USB:PIDerror....2175

SEARch:TRIGger:USB:PMAX....2175
SEARch:TRIGger:USB:PMIN....2175
SEARch:TRIGger:USB:SCONdition....2176
SEARch:TRIGger:USB:SDATa....2177
SEARch:TRIGger:USB:SERRor....2177
SEARch:TRIGger:USB:SHANdshake....2177
SEARch:TRIGger:USB:SMAX....2176
SEARch:TRIGger:USB:SMIN....2176
SEARch:TRIGger:USB:SPEC....2178
SEARch:TRIGger:USB:SSOP....2177
SEARch:TRIGger:USB:SSPE....2178
SEARch:TRIGger:USB:STCO....2178
SEARch:TRIGger:USB:STOKen....2179
SEARch:TRIGger:USB:TCONdition....2179
SEARch:TRIGger:USB:TMAX....2180
SEARch:TRIGger:USB:TMIN....2179
SEARch:TRIGger:USB:TOKen....2180
SEARch:TRIGger:USB:WADD....2180
SEARch:TRIGger:USB:WEND....2181
SEARch:TRIGger:USB:WETCheck....2181
SEARch:TRIGger:USB:WFRN....2181
SEARch:TRIGger:USB:WPAY....2181
SEARch:TRIGger:USB:WPID....2182
SEARch:TRIGger:USB:WPOR....2182
SEARch:TRIGger:USB:WSEU....2182
SEARch:TRIGger:USB:WSTC....2182
SEARch:TRIGger:USBPd:BIT....2234
SEARch:TRIGger:USBPd:DMAX....2234
SEARch:TRIGger:USBPd:DMIN....2235
SEARch:TRIGger:USBPd:DOPerator....2235
SEARch:TRIGger:USBPd:ERENable....2232
SEARch:TRIGger:USBPd:ERRor<m>:ENABle....2232
SEARch:TRIGger:USBPd:FIENable....2233
SEARch:TRIGger:USBPd:FRAMe<m>:ENABle....2233
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:BIT....2234
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMAX....2234
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DMIN....2235
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:DOPerator....2235
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:ENABle....2233
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMAX....2236
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IMIN....2236
SEARch:TRIGger:USBPd:FRAMe<m>:FLD<n>:IOPerator....2237
SEARch:TRIGger:USBPd:FRENable....2233
SEARch:TRIGger:USBPd:IMAX....2236
SEARch:TRIGger:USBPd:IMIN....2236
SEARch:TRIGger:USBPd:IOPerator....2237
SEARch:TRIGger:USBThree:BIT....2212
SEARch:TRIGger:USBThree:DMAX....2211
SEARch:TRIGger:USBThree:DMIN....2211
SEARch:TRIGger:USBThree:DOPerator....2210

SEARch:TRIGger:USBThree:ERENable....2209
SEARch:TRIGger:USBThree:ERRor<m>:ENABle....2209
SEARch:TRIGger:USBThree:FIENable....2210
SEARch:TRIGger:USBThree:FRAMe<m>:ENABle....2209
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:BIT....2212
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DMAX....2211
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DMIN....2211
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:DOPerator....2210
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:ENABle....2210
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMAX....2213
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IMIN....2213
SEARch:TRIGger:USBThree:FRAMe<m>:FLD<n>:IOPerator....2212
SEARch:TRIGger:USBThree:FRENable....2209
SEARch:TRIGger:USBThree:IMAX....2213
SEARch:TRIGger:USBThree:IMIN....2213
SEARch:TRIGger:USBThree:IOPerator....2212
SEARch:TRIGger:WIDTh:ACOPy....1608
SEARch:TRIGger:WIDTh:DELTa....1617
SEARch:TRIGger:WIDTh:POLarity....1617
SEARch:TRIGger:WIDTh:RANGe....1618
SEARch:TRIGger:WIDTh:WIDTh....1618
SEARch:TRIGger:WIDTh[:STATe]....1607
SEARch:TRIGger:WINDow:ACOPy....1608
SEARch:TRIGger:WINDow:DELTa....1619
SEARch:TRIGger:WINDow:RANGe....1619
SEARch:TRIGger:WINDow:TIMerange....1620
SEARch:TRIGger:WINDow:WIDTh....1620
SEARch:TRIGger:WINDow[:STATe]....1607
SENSe[:ROSCillator]:EXTernal:FREQuency....1388
SENSe[:ROSCillator]:SOURce....1388
SINGle....1325
STATus:OPERation:CONDition?....2572
STATus:OPERation:ENABle....2573
STATus:OPERation[:EVENt]?....2572
STATus:QUEStionable:ADCState:CONDition?....2575
STATus:QUEStionable:ADCState:ENABle....2576
STATus:QUEStionable:ADCState:NTRansition....2576
STATus:QUEStionable:ADCState:PTRansition....2577
STATus:QUEStionable:ADCState[:EVENt]?....2576
STATus:QUEStionable:COVerload:CONDition?....2575
STATus:QUEStionable:COVerload:ENABle....2576
STATus:QUEStionable:COVerload:NTRansition....2576
STATus:QUEStionable:COVerload:PTRansition....2577
STATus:QUEStionable:COVerload[:EVENt]?....2576
STATus:QUEStionable:DEEMbedding:CONDition?....2575
STATus:QUEStionable:DEEMbedding:NTRansition....2577
STATus:QUEStionable:DEEMbedding:PTRansition....2577
STATus:QUEStionable:DEEMbedding[:EVENt]?....2576
STATus:QUEStionable:FREQuency:CONDition?....2575
STATus:QUEStionable:FREQuency:ENABle....2576

STATus:QUEStionable:FREQuency:EVENt?.... 2576
STATus:QUEStionable:FREQuency:NTRansition.... 2577
STATus:QUEStionable:FREQuency:PTRansition.... 2577
STATus:QUEStionable:LAMPlitude:CONDition?.... 2575
STATus:QUEStionable:LAMPlitude:ENABle.... 2576
STATus:QUEStionable:LAMPlitude:NTRansition.... 2577
STATus:QUEStionable:LAMPlitude:PTRansition.... 2577
STATus:QUEStionable:LAMPlitude[:EVENt]?.... 2576
STATus:QUEStionable:LIMit:CONDition?.... 2575
STATus:QUEStionable:LIMit:ENABle.... 2576
STATus:QUEStionable:LIMit:NTRansition.... 2577
STATus:QUEStionable:LIMit:PTRansition.... 2577
STATus:QUEStionable:LIMit[:EVENt]?.... 2576
STATus:QUEStionable:MARGin:CONDition?.... 2575
STATus:QUEStionable:MARGin:ENABle.... 2576
STATus:QUEStionable:MARGin:NTRansition.... 2577
STATus:QUEStionable:MARGin:PTRansition.... 2577
STATus:QUEStionable:MARGin[:EVENt]?.... 2576
STATus:QUEStionable:MASK:CONDition?.... 2576
STATus:QUEStionable:MASK:ENABle.... 2576
STATus:QUEStionable:MASK:NTRansition.... 2577
STATus:QUEStionable:MASK:PTRansition.... 2577
STATus:QUEStionable:MASK[:EVENt]?.... 2576
STATus:QUEStionable:TEMPerature:CONDition?.... 2575
STATus:QUEStionable:TEMPerature:ENABle.... 2576
STATus:QUEStionable:TEMPerature:NTRansition.... 2576
STATus:QUEStionable:TEMPerature:PTRansition.... 2577
STATus:QUEStionable:TEMPerature[:EVENt]?.... 2576
STATus:QUEStionable:ZVCoverload:CONDition?.... 2576
STATus:QUEStionable:ZVCoverload:ENABle.... 2576
STATus:QUEStionable:ZVCoverload:NTRansition.... 2577
STATus:QUEStionable:ZVCoverload:PTRansition.... 2577
STATus:QUEStionable:ZVCoverload[:EVENt]?.... 2576
STOP.... 1326
SWTRigger:HISTory.... 1433
SYSTem:DATE.... 1296
SYSTem:DEVice:ID?.... 1297
SYSTem:DFPRint.... 1323
SYSTem:DISPlay:MESSage:STATe.... 1295
SYSTem:DISPlay:MESSage[:TEXT].... 1295
SYSTem:DISPlay:UPDate.... 1294
SYSTem:KLOCk.... 1294
SYSTem:LANGuage.... 1295
SYSTem:PRESet.... 1296
SYSTem:TIME?.... 1296
TDRT:BW.... 2469
TDRT:CALibration:OPEN:NAME.... 2466
TDRT:CALibration:OPEN[:EXECute].... 2466
TDRT:CALibration:SAVE:NAME.... 2468
TDRT:CALibration:SAVE[:EXECute].... 2468

TDRT:CALibration:USECaldata.....2468
TDRT:DEFault.....2466
TDRT:DOMain.....2469
TDRT:PERMittivity.....2469
TDRT:PULSe:OUT.....2467
TDRT:PULSe:REFerence.....2467
TDRT:SIGNal<m>:DATA:HEADer?.....2471
TDRT:SIGNal<m>:DATA:VALues?.....2471
TDRT:SIGNal<m>:OFFSet.....2470
TDRT:SIGNal<m>:POSition.....2470
TDRT:SIGNal<m>:SCALe.....2470
TDRT:SIGNal<m>STATe.....2469
TDRT:SLENgth.....2467
TDRT:STATe.....2468
TDRT:TYPE.....2467
TIMebase:DIVisions?.....1327
TIMebase:HORizontal:POSition.....1327
TIMebase:RACTime?.....1573
TIMebase:RANGe.....1327
TIMebase:REFerence.....1327
TIMebase:ROLL:ENABle.....1331
TIMebase:ROLL:MTIMe.....1331
TIMebase:ROLL:STATe?.....1331
TIMebase:SCALe.....1326
TRACe:REMote:MODE:FILE:DEXecution:DURation.....2581
TRACe:REMote:MODE:FILE:ENABle.....2580
TRACe:REMote:MODE:FILE:FILTer.....2580
TRACe:REMote:MODE:FILE:FORMat.....2579
TRACe:REMote:MODE:FILE:FUNCtions.....2581
TRACe:REMote:MODE:FILE:NAME.....2579
TRACe:REMote:MODE:FILE:PARSer.....2581
TRACe:REMote:MODE:FILE:RPC.....2581
TRACe:REMote:MODE:FILE:SIZE.....2579
TRACe:REMote:MODE:FILE:STARtmode.....2579
TRACe:REMote:MODE:FILE:STOPmode.....2579
TRIGger<m>:ANEDge:COUPling.....1434
TRIGger<m>:ANEDge:CUToff:HIGHpass.....1435
TRIGger<m>:ANEDge:CUToff:LOWPass.....1435
TRIGger<m>:ANEDge:FILTer.....1434
TRIGger<m>:ANEDge:GND.....1436
TRIGger<m>:ANEDge:SLOPe.....1436
TRIGger<m>:ARINc:DATA:CONDition.....1986
TRIGger<m>:ARINc:DATA:MAX.....1987
TRIGger<m>:ARINc:DATA:MIN.....1987
TRIGger<m>:ARINc:ERRor:CODing.....1987
TRIGger<m>:ARINc:ERRor:PARity.....1987
TRIGger<m>:ARINc:LABel:CONDition.....1988
TRIGger<m>:ARINc:LABel:MAX.....1988
TRIGger<m>:ARINc:LABel:MIN.....1988
TRIGger<m>:ARINc:MAXGap:BITS.....1989

TRIGger<m>:ARINc:MAXGap:SELect.... 1989
TRIGger<m>:ARINc:MINGap:BITS.... 1988
TRIGger<m>:ARINc:MINGap:SELect.... 1988
TRIGger<m>:ARINc:SDI.... 1989
TRIGger<m>:ARINc:SSM.... 1989
TRIGger<m>:ARINc:TYPE.... 1986
TRIGger<m>:CAN:ACKerror.... 1807
TRIGger<m>:CAN:BITSterror.... 1807
TRIGger<m>:CAN:BORDer.... 1805
TRIGger<m>:CAN:CRCerror.... 1807
TRIGger<m>:CAN:DCONdition.... 1804
TRIGger<m>:CAN:DLC.... 1805
TRIGger<m>:CAN:DLCCondition.... 1805
TRIGger<m>:CAN:DMAX.... 1805
TRIGger<m>:CAN:DMIN.... 1804
TRIGger<m>:CAN:FDATa:BRS.... 1804
TRIGger<m>:CAN:FDATa:DPOPerator.... 1806
TRIGger<m>:CAN:FDATa:DPOSition.... 1806
TRIGger<m>:CAN:FDATa:DPTO.... 1807
TRIGger<m>:CAN:FDATa:ESI.... 1804
TRIGger<m>:CAN:FDATa:FDF.... 1803
TRIGger<m>:CAN:FDATa:SCERror.... 1808
TRIGger<m>:CAN:FDATa:STANdard.... 1802
TRIGger<m>:CAN:FORMerror.... 1807
TRIGger<m>:CAN:FTYPe.... 1802
TRIGger<m>:CAN:ICONdition.... 1803
TRIGger<m>:CAN:IMAX.... 1803
TRIGger<m>:CAN:IMIN.... 1803
TRIGger<m>:CAN:ITYPe.... 1802
TRIGger<m>:CAN:NDBYtes?.... 1806
TRIGger<m>:CAN:SYMBolic:DMAX.... 1833
TRIGger<m>:CAN:SYMBolic:DMIN.... 1833
TRIGger<m>:CAN:SYMBolic:MSGValue.... 1832
TRIGger<m>:CAN:SYMBolic:SGEValue.... 1833
TRIGger<m>:CAN:SYMBolic:SIGValue.... 1832
TRIGger<m>:CAN:SYMBolic:TSIGnals.... 1832
TRIGger<m>:CAN:TYPE.... 1801
TRIGger<m>:CMSB:ADVanced:BIT.... 1771
TRIGger<m>:CMSB:ADVanced:DMAX.... 1771
TRIGger<m>:CMSB:ADVanced:DMIN.... 1772
TRIGger<m>:CMSB:ADVanced:DOPerator.... 1772
TRIGger<m>:CMSB:ADVanced:ERENable.... 1772
TRIGger<m>:CMSB:ADVanced:ERRor<n>:ENABle.... 1772
TRIGger<m>:CMSB:ADVanced:FIENable.... 1770
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:ENABle.... 1770
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:BIT.... 1771
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMAX.... 1771
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DMIN.... 1772
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:DOPerator.... 1772
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:ENABle.... 1770

TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMAX ... 1773
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IMIN ... 1773
TRIGger<m>:CMSB:ADVanced:FRAMe<n>:FLD<o>:IOPerator ... 1773
TRIGger<m>:CMSB:ADVanced:FRENable ... 1770
TRIGger<m>:CMSB:ADVanced:IMAX ... 1773
TRIGger<m>:CMSB:ADVanced:IMIN ... 1773
TRIGger<m>:CMSB:ADVanced:IOPerator ... 1773
TRIGger<m>:CMSB:ICONdition ... 1769
TRIGger<m>:CMSB:IMAX ... 1770
TRIGger<m>:CMSB:IMIN ... 1769
TRIGger<m>:CMSB:NRZ:WRDLength ... 1770
TRIGger<m>:CMSB:PATTern ... 1769
TRIGger<m>:CMSB:TYPE ... 1768
TRIGger<m>:COUPling ... 1383
TRIGger<m>:CXPI:CT ... 1934
TRIGger<m>:CXPI:DATA:DCONdition ... 1934
TRIGger<m>:CXPI:DATA:DMAX ... 1934
TRIGger<m>:CXPI:DATA:DMIN ... 1935
TRIGger<m>:CXPI:DATA:ICONdition ... 1935
TRIGger<m>:CXPI:DATA:IMAX ... 1935
TRIGger<m>:CXPI:DATA:IMIN ... 1935
TRIGger<m>:CXPI:DEXTension:CONDition ... 1936
TRIGger<m>:CXPI:DEXTension:MAX ... 1936
TRIGger<m>:CXPI:DEXTension:MIN ... 1936
TRIGger<m>:CXPI:DLC:CONDition ... 1937
TRIGger<m>:CXPI:DLC:MAX ... 1937
TRIGger<m>:CXPI:DLC:MIN ... 1937
TRIGger<m>:CXPI:ERRor:CRC ... 1937
TRIGger<m>:CXPI:ERRor:DLC ... 1938
TRIGger<m>:CXPI:ERRor:IBS ... 1938
TRIGger<m>:CXPI:ERRor:IFS ... 1938
TRIGger<m>:CXPI:ERRor:LENGth ... 1938
TRIGger<m>:CXPI:ERRor:PARity ... 1939
TRIGger<m>:CXPI:ERRor:UART ... 1939
TRIGger<m>:CXPI:FID:CONDition ... 1939
TRIGger<m>:CXPI:FID:MAX ... 1940
TRIGger<m>:CXPI:FID:MIN ... 1940
TRIGger<m>:CXPI:NM ... 1940
TRIGger<m>:CXPI:TYPE ... 1940
TRIGger<m>:DPHY:DATA:CONDition ... 2249
TRIGger<m>:DPHY:DATA:MAX ... 2250
TRIGger<m>:DPHY:DATA:MIN ... 2250
TRIGger<m>:DPHY:DIDX:CONDition ... 2250
TRIGger<m>:DPHY:DIDX:MAX ... 2250
TRIGger<m>:DPHY:DIDX:MIN ... 2251
TRIGger<m>:DPHY:DTYPe:CONDition ... 2251
TRIGger<m>:DPHY:DTYPe:MAX ... 2251
TRIGger<m>:DPHY:DTYPe:MIN ... 2251
TRIGger<m>:DPHY:ESCMode:CONDition ... 2252
TRIGger<m>:DPHY:ESCMode:MAX ... 2252

TRIGger<m>:DPHY:ESCMode:MIN ... 2252
TRIGger<m>:DPHY:ESDTa:CONDition ... 2252
TRIGger<m>:DPHY:ESDTa:MAX ... 2253
TRIGger<m>:DPHY:ESDTa:MIN ... 2253
TRIGger<m>:DPHY:ESINdex:CONDition ... 2253
TRIGger<m>:DPHY:ESINdex:MAX ... 2254
TRIGger<m>:DPHY:ESINdex:MIN ... 2254
TRIGger<m>:DPHY:HSVC ... 2254
TRIGger<m>:DPHY:TYPE ... 2249
TRIGger<m>:DPHY:WCOunt:CONDition ... 2254
TRIGger<m>:DPHY:WCOunt:MAX ... 2254
TRIGger<m>:DPHY:WCOunt:MIN ... 2255
TRIGger<m>:EBTB:DISParityerr ... 2336
TRIGger<m>:EBTB:DX ... 2337
TRIGger<m>:EBTB:DY ... 2337
TRIGger<m>:EBTB:EBPA ... 2337
TRIGger<m>:EBTB:GLITcherror ... 2337
TRIGger<m>:EBTB:SSTYpe ... 2337
TRIGger<m>:EBTB:STYPe ... 2338
TRIGger<m>:EBTB:SYME ... 2338
TRIGger<m>:EBTB:SYMType ... 2338
TRIGger<m>:EBTB:TBPA ... 2338
TRIGger<m>:EBTB:UNK ... 2338
TRIGger<m>:ECOupling ... 1428
TRIGger<m>:EDGE:SLOPe ... 1392
TRIGger<m>:ETHernet:ERRor:CRC ... 2020
TRIGger<m>:ETHernet:ERRor:LENGth ... 2021
TRIGger<m>:ETHernet:ERRor:PREamble ... 2021
TRIGger<m>:ETHernet:FRAMe:CCONdition ... 2021
TRIGger<m>:ETHernet:FRAMe:CMAX ... 2022
TRIGger<m>:ETHernet:FRAMe:CMIN ... 2022
TRIGger<m>:ETHernet:FRAMe:DCONdition ... 2022
TRIGger<m>:ETHernet:FRAMe:DMAX ... 2022
TRIGger<m>:ETHernet:FRAMe:DMIN ... 2023
TRIGger<m>:ETHernet:FRAMe:SCONdition ... 2023
TRIGger<m>:ETHernet:FRAMe:SMAX ... 2023
TRIGger<m>:ETHernet:FRAMe:SMIN ... 2023
TRIGger<m>:ETHernet:FRAMe:TCONdition ... 2024
TRIGger<m>:ETHernet:FRAMe:TMAX ... 2024
TRIGger<m>:ETHernet:FRAMe:TMIN ... 2024
TRIGger<m>:ETHernet:PATTern ... 2020
TRIGger<m>:ETHernet:TYPE ... 2020
TRIGger<m>:EVENt:BEEP ... 1431
TRIGger<m>:EVENt:PRINt ... 1432
TRIGger<m>:EVENt:RUNexec ... 1432
TRIGger<m>:EVENt:WFMSave ... 1432
TRIGger<m>:EVENt[:EVENt] ... 1427
TRIGger<m>:EXTern:OVERload ... 1436
TRIGger<m>:FINDlevel ... 1392
TRIGger<m>:FLXRay:BSSerror ... 1877

TRIGger<m>:FLXRay:CENable....1874
TRIGger<m>:FLXRay:CMAX....1875
TRIGger<m>:FLXRay:CMIN....1874
TRIGger<m>:FLXRay:CSTep....1875
TRIGger<m>:FLXRay:DCONdition....1876
TRIGger<m>:FLXRay:DMAX....1876
TRIGger<m>:FLXRay:DMIN....1876
TRIGger<m>:FLXRay:DPOPerator....1875
TRIGger<m>:FLXRay:DPOSition....1875
TRIGger<m>:FLXRay:DPTO....1876
TRIGger<m>:FLXRay:FCONdition....1872
TRIGger<m>:FLXRay:FESerror....1877
TRIGger<m>:FLXRay:FMAX....1873
TRIGger<m>:FLXRay:FMIN....1873
TRIGger<m>:FLXRay:FSSerror....1877
TRIGger<m>:FLXRay:HCRCerror....1878
TRIGger<m>:FLXRay:NUFRame....1872
TRIGger<m>:FLXRay:PCONdition....1873
TRIGger<m>:FLXRay:PCRCerror....1878
TRIGger<m>:FLXRay:PLPReamble....1872
TRIGger<m>:FLXRay:PMAX....1874
TRIGger<m>:FLXRay:PMIN....1874
TRIGger<m>:FLXRay:STFRame....1872
TRIGger<m>:FLXRay:SYFRame....1872
TRIGger<m>:FLXRay:SYMBol....1877
TRIGger<m>:FLXRay:TYPE....1871
TRIGger<m>:FORCe....1430
TRIGger<m>:GLITch:POLarity....1393
TRIGger<m>:GLITch:RANGe....1393
TRIGger<m>:GLITch:WIDTh....1394
TRIGger<m>:HBTO:CRC:CONDition....2094
TRIGger<m>:HBTO:CRC:MAX....2094
TRIGger<m>:HBTO:CRC:MIN....2094
TRIGger<m>:HBTO:DADDress:CONDition....2091
TRIGger<m>:HBTO:DADDress:MAX....2092
TRIGger<m>:HBTO:DADDress:MIN....2091
TRIGger<m>:HBTO:DATA:DCONdition....2095
TRIGger<m>:HBTO:DATA:DMAX....2095
TRIGger<m>:HBTO:DATA:DMIN....2095
TRIGger<m>:HBTO:DATA:ICONdition....2096
TRIGger<m>:HBTO:DATA:IMAX....2096
TRIGger<m>:HBTO:DATA:IMIN....2096
TRIGger<m>:HBTO:ERRor:CRC....2097
TRIGger<m>:HBTO:ERRor:PREamble....2096
TRIGger<m>:HBTO:ERRor:SFD....2097
TRIGger<m>:HBTO:LENGth:CONDition....2093
TRIGger<m>:HBTO:LENGth:MAX....2093
TRIGger<m>:HBTO:LENGth:MIN....2093
TRIGger<m>:HBTO:SADDress:CONDition....2092
TRIGger<m>:HBTO:SADDress:MAX....2093

TRIGger<m>:HBTO:SADDress:MIN....2092
TRIGger<m>:HBTO:TYPE....2090
TRIGger<m>:HOLDoff:AUTotime?.... 1423
TRIGger<m>:HOLDoff:EVENts.... 1422
TRIGger<m>:HOLDoff:MAX.... 1423
TRIGger<m>:HOLDoff:MIN.... 1422
TRIGger<m>:HOLDoff:MODE.... 1420
TRIGger<m>:HOLDoff:SCALing....1423
TRIGger<m>:HOLDoff:TIME.... 1421
TRIGger<m>:I2C:ACCess.... 1679
TRIGger<m>:I2C:ACONdition.... 1681
TRIGger<m>:I2C:ADDRess....1681
TRIGger<m>:I2C:ADDTo....1681
TRIGger<m>:I2C:ADNack.... 1680
TRIGger<m>:I2C:ADOR<n>:ADRType.... 1682
TRIGger<m>:I2C:ADOR<n>:ENABle.... 1681
TRIGger<m>:I2C:ADOR<n>:RWBit....1682
TRIGger<m>:I2C:ADOR<n>[:VALue].... 1682
TRIGger<m>:I2C:AMODe....1680
TRIGger<m>:I2C:DCONdition.... 1683
TRIGger<m>:I2C:DMAX.... 1684
TRIGger<m>:I2C:DMIN.... 1684
TRIGger<m>:I2C:DPOPerator....1683
TRIGger<m>:I2C:DPOSition....1683
TRIGger<m>:I2C:DPTO.... 1683
TRIGger<m>:I2C:DRNack....1680
TRIGger<m>:I2C:DWNack.... 1680
TRIGger<m>:I2C:TYPE....1679
TRIGger<m>:I2S:SOWords....1746
TRIGger<m>:I2S:TCONdition<n>:CHANnel....1744
TRIGger<m>:I2S:TCONdition<n>:CONDtion.... 1745
TRIGger<m>:I2S:TCONdition<n>:DMAX.... 1746
TRIGger<m>:I2S:TCONdition<n>:DMIN.... 1746
TRIGger<m>:I2S:TYPE.... 1743
TRIGger<m>:I2S:WSSLope.... 1747
TRIGger<m>:INTerval:DELTa.... 1402
TRIGger<m>:INTerval:RANGe.... 1402
TRIGger<m>:INTerval:SLOPe.... 1401
TRIGger<m>:INTerval:WIDTh....1402
TRIGger<m>:LEVel<n>:NOISe:ABSolute....1425
TRIGger<m>:LEVel<n>:NOISe:MODE....1425
TRIGger<m>:LEVel<n>:NOISe:PERDivision....1426
TRIGger<m>:LEVel<n>:NOISe:RELative....1426
TRIGger<m>:LEVel<n>:NOISe[:STATe].... 1424
TRIGger<m>:LEVel<n>:RUNT:LOWer.... 1396
TRIGger<m>:LEVel<n>:RUNT:UPPer.... 1396
TRIGger<m>:LEVel<n>:SLEW:LOWer....1403
TRIGger<m>:LEVel<n>:SLEW:UPPer....1403
TRIGger<m>:LEVel<n>:WINDow:LOWer....1398
TRIGger<m>:LEVel<n>:WINDow:UPPer....1398

TRIGger<m>:LEVel<n>[:VALue]....1391
TRIGger<m>:LEVel<n>[:VALue]....2518
TRIGger<m>:LIN:BORDer....1842
TRIGger<m>:LIN:CHKSerror....1844
TRIGger<m>:LIN:CRCDatalen....1844
TRIGger<m>:LIN:DCONdition....1841
TRIGger<m>:LIN:DLECondition....1842
TRIGger<m>:LIN:DLENgth....1842
TRIGger<m>:LIN:DMAX....1841
TRIGger<m>:LIN:DMIN....1841
TRIGger<m>:LIN:ERRPattern....1844
TRIGger<m>:LIN:ICONdition....1840
TRIGger<m>:LIN:IDOR<n>:ENABle....1843
TRIGger<m>:LIN:IDOR<n>[:VALue]....1843
TRIGger<m>:LIN:IMAX....1841
TRIGger<m>:LIN:IMIN....1841
TRIGger<m>:LIN:IPERror....1843
TRIGger<m>:LIN:STANdard....1845
TRIGger<m>:LIN:SYERror....1843
TRIGger<m>:LIN:TYPE....1839
TRIGger<m>:MDIO:DATA....2045
TRIGger<m>:MDIO:FRAMetype....2044
TRIGger<m>:MDIO:PHYS....2044
TRIGger<m>:MDIO:REGI....2044
TRIGger<m>:MDIO:ST....2043
TRIGger<m>:MDIO:TYPE....2043
TRIGger<m>:MILStd:CDST:ICONdition....1964
TRIGger<m>:MILStd:CDST:IMAX....1965
TRIGger<m>:MILStd:CDST:IMIN....1965
TRIGger<m>:MILStd:CDST:RCONdition....1961
TRIGger<m>:MILStd:CDST:RMAX....1962
TRIGger<m>:MILStd:CDST:RMIN....1962
TRIGger<m>:MILStd:CMD:CCONdition....1962
TRIGger<m>:MILStd:CMD:CMAX....1963
TRIGger<m>:MILStd:CMD:CMIN....1963
TRIGger<m>:MILStd:CMD:RCONdition....1961
TRIGger<m>:MILStd:CMD:RMAX....1962
TRIGger<m>:MILStd:CMD:RMIN....1962
TRIGger<m>:MILStd:CMD:SCONdition....1963
TRIGger<m>:MILStd:CMD:SMAX....1963
TRIGger<m>:MILStd:CMD:SMIN....1964
TRIGger<m>:MILStd:CMD:TR....1964
TRIGger<m>:MILStd:DATA:DCONdition....1965
TRIGger<m>:MILStd:DATA:DMAX....1965
TRIGger<m>:MILStd:DATA:DMIN....1965
TRIGger<m>:MILStd:DATA:ICONdition....1966
TRIGger<m>:MILStd:DATA:IMAX....1966
TRIGger<m>:MILStd:DATA:IMIN....1966
TRIGger<m>:MILStd:DATA:RCONdition....1961
TRIGger<m>:MILStd:DATA:RMAX....1962

TRIGger<m>:MILStd:DATA:RMIN.... 1962
TRIGger<m>:MILStd:ERRor:MANChester.... 1966
TRIGger<m>:MILStd:ERRor:PARity.... 1967
TRIGger<m>:MILStd:ERRor:SYNC.... 1967
TRIGger<m>:MILStd:MAXResponse:BITS.... 1967
TRIGger<m>:MILStd:MAXResponse:SELect.... 1967
TRIGger<m>:MILStd:MINGap:BITS.... 1968
TRIGger<m>:MILStd:MINGap:SELect.... 1968
TRIGger<m>:MILStd:STATus:BCReceived.... 1968
TRIGger<m>:MILStd:STATus:BUSY.... 1968
TRIGger<m>:MILStd:STATus:DBCaccept.... 1968
TRIGger<m>:MILStd:STATus:INSTrument.... 1969
TRIGger<m>:MILStd:STATus:MERRor.... 1969
TRIGger<m>:MILStd:STATus:SREQuest.... 1969
TRIGger<m>:MILStd:STATus:SUBSystem.... 1969
TRIGger<m>:MILStd:STATus:TERMinal.... 1969
TRIGger<m>:MILStd:TPSPecifier.... 1969
TRIGger<m>:MILStd:TYPE.... 1961
TRIGger<m>:MODE.... 1430
TRIGger<m>:MPHY:CRC:CONDition.... 2286
TRIGger<m>:MPHY:CRC:MAX.... 2287
TRIGger<m>:MPHY:CRC:MIN.... 2287
TRIGger<m>:MPHY:CREDit:CONDition.... 2288
TRIGger<m>:MPHY:CREDit:MAX.... 2289
TRIGger<m>:MPHY:CREDit:MIN.... 2289
TRIGger<m>:MPHY:CREQ.... 2287
TRIGger<m>:MPHY:DATA:DCON.... 2284
TRIGger<m>:MPHY:DATA:DMAX.... 2284
TRIGger<m>:MPHY:DATA:DMIN.... 2284
TRIGger<m>:MPHY:DATA:ICONdition.... 2285
TRIGger<m>:MPHY:DATA:IMAX.... 2285
TRIGger<m>:MPHY:DATA:IMIN.... 2285
TRIGger<m>:MPHY:ERRor:LCMD.... 2291
TRIGger<m>:MPHY:ERRor:LENGth.... 2291
TRIGger<m>:MPHY:ERRor:REServed.... 2292
TRIGger<m>:MPHY:ERRor:SYMBol.... 2292
TRIGger<m>:MPHY:ERRor:UNKNown.... 2292
TRIGger<m>:MPHY:FSNumber:CONDition.... 2285
TRIGger<m>:MPHY:FSNumber:MAX.... 2286
TRIGger<m>:MPHY:FSNumber:MIN.... 2286
TRIGger<m>:MPHY:LCCType.... 2279
TRIGger<m>:MPHY:LWFour:CONDition.... 2282
TRIGger<m>:MPHY:LWFour:MAX.... 2283
TRIGger<m>:MPHY:LWFour:MIN.... 2283
TRIGger<m>:MPHY:LWONe:CONDition.... 2279
TRIGger<m>:MPHY:LWONe:MAX.... 2280
TRIGger<m>:MPHY:LWONe:MIN.... 2280
TRIGger<m>:MPHY:LWTHree:CONDition.... 2281
TRIGger<m>:MPHY:LWTHree:MAX.... 2282
TRIGger<m>:MPHY:LWTHree:MIN.... 2282

TRIGger<m>:MPHY:LWTWo:CONDition....2280
TRIGger<m>:MPHY:LWTWo:MAX....2281
TRIGger<m>:MPHY:LWTWo:MIN....2281
TRIGger<m>:MPHY:PBEGin:CONDition....2289
TRIGger<m>:MPHY:PBEGin:MAX....2290
TRIGger<m>:MPHY:PBEGin:MIN....2289
TRIGger<m>:MPHY:PFID:CONDition....2290
TRIGger<m>:MPHY:PFID:MAX....2291
TRIGger<m>:MPHY:PFID:MIN....2290
TRIGger<m>:MPHY:PRPLength....2291
TRIGger<m>:MPHY:RREQ....2288
TRIGger<m>:MPHY:SSIC:BIT....2322
TRIGger<m>:MPHY:SSIC:DMAX....2323
TRIGger<m>:MPHY:SSIC:DMIN....2323
TRIGger<m>:MPHY:SSIC:DOPerator....2323
TRIGger<m>:MPHY:SSIC:ERENable....2321
TRIGger<m>:MPHY:SSIC:ERRor<n>:ENABle....2321
TRIGger<m>:MPHY:SSIC:FIENable....2322
TRIGger<m>:MPHY:SSIC:FRAMe<n>:ENABle....2321
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:BIT....2322
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DMAX....2323
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DMIN....2323
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:DOPerator....2323
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:ENABle....2322
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IMAX....2324
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IMIN....2324
TRIGger<m>:MPHY:SSIC:FRAMe<n>:FLD<o>:IOPerator....2325
TRIGger<m>:MPHY:SSIC:FRENable....2321
TRIGger<m>:MPHY:SSIC:IMAX....2324
TRIGger<m>:MPHY:SSIC:IMIN....2324
TRIGger<m>:MPHY:SSIC:IOPerator....2325
TRIGger<m>:MPHY:TC....2283
TRIGger<m>:MPHY:TYPE....2278
TRIGger<m>:NFC:BITRate....2478
TRIGger<m>:NFC:EVENt....2478
TRIGger<m>:NFC:TECHnology....2478
TRIGger<m>:OFFSet:LIMited....1328
TRIGger<m>:OUT:ACTion....1433
TRIGger<m>:OUT:DELay....1431
TRIGger<m>:OUT:PLENgth....1431
TRIGger<m>:OUT:POLarity....1430
TRIGger<m>:PARallel:EDGE:EXPRession[:DEFine]....2433
TRIGger<m>:PARallel:EDGE:SLOPe....2433
TRIGger<m>:PARallel:PATTern:BIT<0..15>....2437
TRIGger<m>:PARallel:PATTern:EXPRession[:DEFine]....2433
TRIGger<m>:PARallel:PATTern:MODE....2438
TRIGger<m>:PARallel:PATTern:TIMeout:MODE....2438
TRIGger<m>:PARallel:PATTern:TIMeout[:TIME]....2438
TRIGger<m>:PARallel:PATTern:WIDTh:DELTa....2440
TRIGger<m>:PARallel:PATTern:WIDTh:RANGe....2439

TRIGger<m>:PARallel:PATTern:WIDTh[:WIDTh]....2439
TRIGger<m>:PARallel:SETHold:CSOurce:EDGE....2436
TRIGger<m>:PARallel:SETHold:CSOurce[:VALue]....2433
TRIGger<m>:PARallel:SETHold:HTIMe....2436
TRIGger<m>:PARallel:SETHold:STIMe....2437
TRIGger<m>:PARallel:SPATtern:CSOurce:EDGE....2440
TRIGger<m>:PARallel:SPATtern:CSOurce[:VALue]....2433
TRIGger<m>:PARallel:SPATtern:EXPRession[:DEFine]....2433
TRIGger<m>:PARallel:SPATtern:PATTern....2440
TRIGger<m>:PARallel:STATe:BIT<0..15>....2437
TRIGger<m>:PARallel:STATe:CSOurce:EDGE....2437
TRIGger<m>:PARallel:STATe:CSOurce:VALue....2433
TRIGger<m>:PARallel:STATe:EXPRession[:DEFine]....2433
TRIGger<m>:PARallel:TIMeout:EXPRession[:DEFine]....2433
TRIGger<m>:PARallel:TIMeout:RANGe....2435
TRIGger<m>:PARallel:TIMeout:TIME....2436
TRIGger<m>:PARallel:TYPE....2432
TRIGger<m>:PARallel:WIDTh:DELTa....2435
TRIGger<m>:PARallel:WIDTh:EXPRession[:DEFine]....2433
TRIGger<m>:PARallel:WIDTh:POLarity....2434
TRIGger<m>:PARallel:WIDTh:RANGe....2434
TRIGger<m>:PARallel:WIDTh:WIDTh....2434
TRIGger<m>:PATTern:MODE....1408
TRIGger<m>:PATTern:TIMeout:MODE....1409
TRIGger<m>:PATTern:TIMeout[:TIME]....1409
TRIGger<m>:PATTern:WIDTh:DELTa....1410
TRIGger<m>:PATTern:WIDTh:RANGe....1409
TRIGger<m>:PATTern:WIDTh[:WIDTh]....1410
TRIGger<m>:PCIE:DLLP:FCTL:CRET....2352
TRIGger<m>:PCIE:DLLP:FCTL:VCID....2352
TRIGger<m>:PCIE:DLLP:MULT:AUTH....2352
TRIGger<m>:PCIE:DLLP:MULT:CRET....2353
TRIGger<m>:PCIE:DLLP:MULT:DVPT....2353
TRIGger<m>:PCIE:DLLP:MULT:HABS....2353
TRIGger<m>:PCIE:DLLP:MULT:MIXT....2353
TRIGger<m>:PCIE:DLLP:MULT:PHAS....2353
TRIGger<m>:PCIE:DLLP:MULT:RESA....2353
TRIGger<m>:PCIE:DLLP:MULT:TLPT....2354
TRIGger<m>:PCIE:DLLP:MULT:TYPE....2354
TRIGger<m>:PCIE:DLLP:MULT:VHFC....2354
TRIGger<m>:PCIE:DLLP:MULT:VHGR....2354
TRIGger<m>:PCIE:DLLP:MULT:VLNR....2355
TRIGger<m>:PCIE:DLLP:POWM....2355
TRIGger<m>:PCIE:DLLP:SEQ....2355
TRIGger<m>:PCIE:DLLP:TYPE....2355
TRIGger<m>:PCIE:DLLP:VPAT....2355
TRIGger<m>:PCIE:ERRC:CRC....2355
TRIGger<m>:PCIE:ERRC:DISP....2356
TRIGger<m>:PCIE:ERRC:ECRC....2356
TRIGger<m>:PCIE:ERRC:INVP....2356

TRIGger<m>:PCIE:ERRC:LCRC....2356
TRIGger<m>:PCIE:OSET:TYPE....2356
TRIGger<m>:PCIE:TLP:ADRT....2357
TRIGger<m>:PCIE:TLP:CFGT....2357
TRIGger<m>:PCIE:TLP:CPID....2357
TRIGger<m>:PCIE:TLP:CPLS....2357
TRIGger<m>:PCIE:TLP:DEID....2357
TRIGger<m>:PCIE:TLP:MERW....2358
TRIGger<m>:PCIE:TLP:MSGC....2358
TRIGger<m>:PCIE:TLP:MSGR....2358
TRIGger<m>:PCIE:TLP:ORDE....2358
TRIGger<m>:PCIE:TLP:REID....2358
TRIGger<m>:PCIE:TLP:SNOO....2358
TRIGger<m>:PCIE:TLP:SNUM....2359
TRIGger<m>:PCIE:TLP:TCHN....2359
TRIGger<m>:PCIE:TLP:TYPE....2359
TRIGger<m>:PCIE:TYPE....2359
TRIGger<m>:QUALify<n>:A:LOGic....1419
TRIGger<m>:QUALify<n>:A[:ENABle]....1418
TRIGger<m>:QUALify<n>:AB:LOGic....1420
TRIGger<m>:QUALify<n>:ABCD:LOGic....1420
TRIGger<m>:QUALify<n>:B:LOGic....1419
TRIGger<m>:QUALify<n>:B[:ENABle]....1418
TRIGger<m>:QUALify<n>:C:LOGic....1419
TRIGger<m>:QUALify<n>:C[:ENABle]....1418
TRIGger<m>:QUALify<n>:CD:LOGic....1420
TRIGger<m>:QUALify<n>:D:LOGic....1419
TRIGger<m>:QUALify<n>:D[:ENABle]....1418
TRIGger<m>:QUALify<n>:STATe....1418
TRIGger<m>:RFFE:ADDRess:CONDition....2063
TRIGger<m>:RFFE:ADDRess:MAX....2064
TRIGger<m>:RFFE:ADDRess:MIN....2063
TRIGger<m>:RFFE:BCOunt:CONDition....2064
TRIGger<m>:RFFE:BCOunt:MAX....2064
TRIGger<m>:RFFE:BCOunt:MIN....2064
TRIGger<m>:RFFE:DATA:DCON....2065
TRIGger<m>:RFFE:DATA:DMAX....2065
TRIGger<m>:RFFE:DATA:DMIN....2065
TRIGger<m>:RFFE:DATA:ICONdition....2065
TRIGger<m>:RFFE:DATA:IMAX....2066
TRIGger<m>:RFFE:DATA:IMIN....2066
TRIGger<m>:RFFE:ERRor:BP....2068
TRIGger<m>:RFFE:ERRor:GAP....2069
TRIGger<m>:RFFE:ERRor:LENGth....2068
TRIGger<m>:RFFE:ERRor:NOResponse....2068
TRIGger<m>:RFFE:ERRor:PARity....2068
TRIGger<m>:RFFE:ERRor:SSC....2068
TRIGger<m>:RFFE:ERRor:USEQuence....2069
TRIGger<m>:RFFE:ERRor:VERSion....2069
TRIGger<m>:RFFE:INTerrupt....2068

TRIGger<m>:RFFE:INTerrupt....2069
TRIGger<m>:RFFE:MASK:CONDition....2067
TRIGger<m>:RFFE:MASK:MAX....2067
TRIGger<m>:RFFE:MASK:MIN....2067
TRIGger<m>:RFFE:SADD:CONDition....2066
TRIGger<m>:RFFE:SADD:MAX....2067
TRIGger<m>:RFFE:SADD:MIN....2067
TRIGger<m>:RFFE:TYPE....2061
TRIGger<m>:RFReject....1384
TRIGger<m>:RUNT:DELTa....1397
TRIGger<m>:RUNT:POLarity....1396
TRIGger<m>:RUNT:RANGe....1396
TRIGger<m>:RUNT:WIDTh....1397
TRIGger<m>:SCOupling....1407
TRIGger<m>:SENT:CRCerror....1908
TRIGger<m>:SENT:FORMerror....1907
TRIGger<m>:SENT:IRFLength....1909
TRIGger<m>:SENT:PPERioderror....1908
TRIGger<m>:SENT:PULSeerror....1908
TRIGger<m>:SENT:SDCN....1907
TRIGger<m>:SENT:SDMN....1907
TRIGger<m>:SENT:SDMX....1907
TRIGger<m>:SENT:SICN....1906
TRIGger<m>:SENT:SIDType....1906
TRIGger<m>:SENT:SIMN....1906
TRIGger<m>:SENT:SIMX....1906
TRIGger<m>:SENT:STATus....1904
TRIGger<m>:SENT:STYPe....1905
TRIGger<m>:SENT:TDCN....1905
TRIGger<m>:SENT:TDMN....1905
TRIGger<m>:SENT:TDMX....1905
TRIGger<m>:SENT:TTYPe....1904
TRIGger<m>:SENT:TYPE....1903
TRIGger<m>:SEQuence:COUNt....1428
TRIGger<m>:SEQuence:DELay....1428
TRIGger<m>:SEQuence:RESet:EVENt....1428
TRIGger<m>:SEQuence:RESet:TIMeout:TIME....1429
TRIGger<m>:SEQuence:RESet:TIMeout[:ENABle]....1429
TRIGger<m>:SEQuence:TYPE....1427
TRIGger<m>:SETHold:CSOurce:EDGE....1405
TRIGger<m>:SETHold:CSOurce:LEVel....1406
TRIGger<m>:SETHold:CSOurce[:VALue]....1406
TRIGger<m>:SETHold:HTIMe....1406
TRIGger<m>:SETHold:STIMe....1407
TRIGger<m>:SLEW:DELTa....1405
TRIGger<m>:SLEW:RANGe....1404
TRIGger<m>:SLEW:RATE....1404
TRIGger<m>:SLEW:SLOPe....1403
TRIGger<m>:SOURce:SBSelect....1675
TRIGger<m>:SOURce[:SELect]....1389

TRIGger<m>:SOURce[:SELect]....1674
TRIGger<m>:SOURce[:SELect]....2431
TRIGger<m>:SPATtern:CDR....2518
TRIGger<m>:SPATtern:CSOurce:EDGE....1411
TRIGger<m>:SPATtern:CSOurce:LEVel....1411
TRIGger<m>:SPATtern:CSOurce[:VALue]....1411
TRIGger<m>:SPATtern:PATTern....1412
TRIGger<m>:SPATtern:REFSource....1412
TRIGger<m>:SPI:DPOPerator....1714
TRIGger<m>:SPI:DPOSition....1714
TRIGger<m>:SPI:DPTO....1714
TRIGger<m>:SPI:FCONdition....1714
TRIGger<m>:SPI:MISopattern....1715
TRIGger<m>:SPI:MOSipattern....1715
TRIGger<m>:SPI:PALignment....1713
TRIGger<m>:SPI:TYPE....1713
TRIGger<m>:SWIRe:CTYPe....2003
TRIGger<m>:SWIRe:DATA:CONDition....2003
TRIGger<m>:SWIRe:DATA:MAX....2003
TRIGger<m>:SWIRe:DATA:MIN....2004
TRIGger<m>:SWIRe:ERRor:ESC....2004
TRIGger<m>:SWIRe:ERRor:PARity....2004
TRIGger<m>:SWIRe:TIME:CONDition....2004
TRIGger<m>:SWIRe:TIME:MAX....2005
TRIGger<m>:SWIRe:TIME:MIN....2005
TRIGger<m>:SWIRe:TYPE....2005
TRIGger<m>:TBTO:CRC:CONDition....2126
TRIGger<m>:TBTO:CRC:MAX....2127
TRIGger<m>:TBTO:CRC:MIN....2127
TRIGger<m>:TBTO:DADDress:CONDition....2127
TRIGger<m>:TBTO:DADDress:MAX....2128
TRIGger<m>:TBTO:DADDress:MIN....2128
TRIGger<m>:TBTO:DATA:DCONdition....2128
TRIGger<m>:TBTO:DATA:DMAX....2129
TRIGger<m>:TBTO:DATA:DMIN....2129
TRIGger<m>:TBTO:DATA:ICONdition....2129
TRIGger<m>:TBTO:DATA:IMAX....2130
TRIGger<m>:TBTO:DATA:IMIN....2130
TRIGger<m>:TBTO:ERRor:CRC....2130
TRIGger<m>:TBTO:ERRor:FEC....2131
TRIGger<m>:TBTO:ERRor:OOR....2131
TRIGger<m>:TBTO:ERRor:ZERO....2131
TRIGger<m>:TBTO:LENGth:CONDition....2131
TRIGger<m>:TBTO:LENGth:MAX....2132
TRIGger<m>:TBTO:LENGth:MIN....2132
TRIGger<m>:TBTO:SADDress:CONDition....2132
TRIGger<m>:TBTO:SADDress:MAX....2133
TRIGger<m>:TBTO:SADDress:MIN....2133
TRIGger<m>:TBTO:TRGLevel....2126
TRIGger<m>:TBTO:TYPE....2126

TRIGger<m>:TIMeout:RANGe ... 1400
TRIGger<m>:TIMeout:TIME ... 1401
TRIGger<m>:TV:CUSTom:LDURation ... 1416
TRIGger<m>:TV:CUSTom:SCANmode ... 1415
TRIGger<m>:TV:CUSTom:SDURation ... 1417
TRIGger<m>:TV:CUSTom:STYPe ... 1416
TRIGger<m>:TV:LFIeld ... 1415
TRIGger<m>:TV:LINE ... 1414
TRIGger<m>:TV:MODE ... 1413
TRIGger<m>:TV:POLarity ... 1414
TRIGger<m>:TV:STANdard ... 1413
TRIGger<m>:TYPE ... 1390
TRIGger<m>:UART:DATA ... 1734
TRIGger<m>:UART:DPOPerator ... 1733
TRIGger<m>:UART:DPOSition ... 1734
TRIGger<m>:UART:DPTO ... 1734
TRIGger<m>:UART:FCONdition ... 1734
TRIGger<m>:UART:SOURce ... 1733
TRIGger<m>:UART:TYPE ... 1733
TRIGger<m>:USB:ACONdition ... 2153
TRIGger<m>:USB:AMAX ... 2153
TRIGger<m>:USB:AMIN ... 2153
TRIGger<m>:USB:DATA ... 2154
TRIGger<m>:USB:DCONdition ... 2154
TRIGger<m>:USB:DPOPerator ... 2154
TRIGger<m>:USB:DPOSition ... 2154
TRIGger<m>:USB:ECONdition ... 2155
TRIGger<m>:USB:EMAX ... 2155
TRIGger<m>:USB:EMIN ... 2155
TRIGger<m>:USB:ERRC ... 2155
TRIGger<m>:USB:FCONdition ... 2156
TRIGger<m>:USB:FMAX ... 2157
TRIGger<m>:USB:FMIN ... 2156
TRIGger<m>:USB:HAND ... 2157
TRIGger<m>:USB:PATT ... 2157
TRIGger<m>:USB:PCONdition ... 2157
TRIGger<m>:USB:PMAX ... 2158
TRIGger<m>:USB:PMIN ... 2158
TRIGger<m>:USB:SCONdition ... 2158
TRIGger<m>:USB:SMAX ... 2159
TRIGger<m>:USB:SMIN ... 2158
TRIGger<m>:USB:SPEC ... 2159
TRIGger<m>:USB:STCO ... 2159
TRIGger<m>:USB:TCONdition ... 2160
TRIGger<m>:USB:TMAX ... 2160
TRIGger<m>:USB:TMIN ... 2160
TRIGger<m>:USB:TOKen ... 2160
TRIGger<m>:USB:TYPE ... 2152
TRIGger<m>:USB:WADD ... 2161
TRIGger<m>:USB:WEND ... 2161

TRIGger<m>:USB:WETCheck 2161
TRIGger<m>:USB:WFRN 2161
TRIGger<m>:USB:WPAY 2161
TRIGger<m>:USB:WPID 2162
TRIGger<m>:USB:WPOR 2162
TRIGger<m>:USB:WSEU 2162
TRIGger<m>:USB:WSTC 2162
TRIGger<m>:USBPd:BIT 2224
TRIGger<m>:USBPd:DMAX 2224
TRIGger<m>:USBPd:DMIN 2225
TRIGger<m>:USBPd:DOPerator 2225
TRIGger<m>:USBPd:ERENable 2222
TRIGger<m>:USBPd:ERRor<n>:ENABle 2222
TRIGger<m>:USBPd:FIENable 2223
TRIGger<m>:USBPd:FRAMe<n>:ENABle 2223
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:BIT 2224
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMAX 2224
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DMIN 2225
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:DOPerator 2225
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:ENABle 2223
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMAX 2225
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IMIN 2226
TRIGger<m>:USBPd:FRAMe<n>:FLD<o>:IOPerator 2226
TRIGger<m>:USBPd:FRENable 2223
TRIGger<m>:USBPd:IMAX 2225
TRIGger<m>:USBPd:IMIN 2226
TRIGger<m>:USBPd:IOPerator 2226
TRIGger<m>:USBThree:BIT 2202
TRIGger<m>:USBThree:DMAX 2202
TRIGger<m>:USBThree:DMIN 2201
TRIGger<m>:USBThree:DOPerator 2201
TRIGger<m>:USBThree:ERENable 2199
TRIGger<m>:USBThree:ERRor<n>:ENABle 2199
TRIGger<m>:USBThree:FIENable 2200
TRIGger<m>:USBThree:FRAMe<n>:ENABle 2200
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:BIT 2202
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DMAX 2202
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DMIN 2201
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:DOPerator 2201
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:ENABle 2200
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IMAX 2203
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IMIN 2203
TRIGger<m>:USBThree:FRAMe<n>:FLD<o>:IOPerator 2202
TRIGger<m>:USBThree:FRENable 2200
TRIGger<m>:USBThree:IMAX 2203
TRIGger<m>:USBThree:IMIN 2203
TRIGger<m>:USBThree:IOPerator 2202
TRIGger<m>:WIDTh:DELTa 1395
TRIGger<m>:WIDTh:POLarity 1394
TRIGger<m>:WIDTh:RANGe 1394

TRIGger<m>:WIDTh:WIDTh....1395
TRIGger<m>:WINDow:DELTa....1400
TRIGger<m>:WINDow:RANGe....1398
TRIGger<m>:WINDow:TIME....1399
TRIGger<m>:WINDow:WIDTh....1400
TRIGger<m>:ZONE:EXPRession[:DEFine]....1433
USRDefined:RST:NAME....1321
USRDefined:RST:OPEN....1321
USRDefined:RST:SAVE....1322
USRDefined:RST[:ENABle]....1322
WAVeform<m>:XYCurve:RATio....1465
WAVeform<m>:XYCurve:STATe....1466
WAVeform<m>:XYCurve:SWAP....1466
WAVeform<m>:XYCurve:XSOurce....1467
WAVeform<m>:XYCurve:YSOurce....1467
WGENerator<m>:ACOPy....2443
WGENerator<m>:ARBGen:COPY....2453
WGENerator<m>:ARBGen:MULTichannel:IMPort....2453
WGENerator<m>:ARBGen:MULTichannel:NAME....2453
WGENerator<m>:ARBGen:MULTichannel:OPEN....2453
WGENerator<m>:ARBGen:NAME....2454
WGENerator<m>:ARBGen:OPEN....2454
WGENerator<m>:ARBGen:RUNMode....2454
WGENerator<m>:ARBGen:RUNSingle....2456
WGENerator<m>:ARBGen:SAMPles?....2454
WGENerator<m>:ARBGen:SELect....2455
WGENerator<m>:ARBGen:SRATe....2455
WGENerator<m>:ARBGen[:SOURce]....2455
WGENerator<m>:COUPling:AMPLitude....2462
WGENerator<m>:COUPling:AWForm....2463
WGENerator<m>:COUPling:NOISe....2463
WGENerator<m>:COUPling:PHASeshift....2462
WGENerator<m>:COUPling:STATe....2462
WGENerator<m>:COUPling[:FREQuency]....2463
WGENerator<m>:FREQuency....2444
WGENerator<m>:FUNCtion:PULSe[:WIDTh]....2445
WGENerator<m>:FUNCtion:RAMP[:SYMMetry]....2445
WGENerator<m>:FUNCtion:SQUare:DCYCle....2445
WGENerator<m>:FUNCtion[:SELect]....2444
WGENerator<m>:MODulation:AM:DCYCle....2446
WGENerator<m>:MODulation:AM:DEPTh....2446
WGENerator<m>:MODulation:AM:FREQuency....2447
WGENerator<m>:MODulation:AM:SYMMetry....2447
WGENerator<m>:MODulation:AM[:FUNCtion]....2448
WGENerator<m>:MODulation:CARRier:FREQuency....2447
WGENerator<m>:MODulation:CARRier:PERiod....2447
WGENerator<m>:MODulation:FM:DCYCle....2448
WGENerator<m>:MODulation:FM:DEViation....2448
WGENerator<m>:MODulation:FM:FREQuency....2448
WGENerator<m>:MODulation:FM:SYMMetry....2449

WGENerator<m>:MODulation:FM[:FUNCtion]....2449
WGENerator<m>:MODulation:FSK:FONE....2449
WGENerator<m>:MODulation:FSK:FTWO....2450
WGENerator<m>:MODulation:FSK[:RATE]....2450
WGENerator<m>:MODulation:NDCLevel....2459
WGENerator<m>:MODulation:NLABsolute?....2460
WGENerator<m>:MODulation:NLPCent....2460
WGENerator<m>:MODulation:NOISe....2460
WGENerator<m>:MODulation:PWM:DCYCle....2450
WGENerator<m>:MODulation:PWM:DEPTh....2450
WGENerator<m>:MODulation:PWM:FREQuency....2451
WGENerator<m>:MODulation:PWM:SYMMetry....2451
WGENerator<m>:MODulation:PWM[:FUNCtion]....2451
WGENerator<m>:MODulation:TYPE....2446
WGENerator<m>:OUTPut[:LOAD]....2457
WGENerator<m>:PERiod....2444
WGENerator<m>:PRESet....2443
WGENerator<m>:SOURce....2443
WGENerator<m>:SWEep:FSTart....2452
WGENerator<m>:SWEep:TIME....2452
WGENerator<m>:SWEep[:FEND]....2452
WGENerator<m>:VOLTage:DCLevel....2458
WGENerator<m>:VOLTage:HIGH....2458
WGENerator<m>:VOLTage:INVersion....2459
WGENerator<m>:VOLTage:LOW....2459
WGENerator<m>:VOLTage:OFFSet....2459
WGENerator<m>:VOLTage[:VPP]....2458
WGENerator<m>[:ENABle]....2443
ZVC:BANDwidth....1367
ZVC:RESCoupled....1368
ZVC:TYPE....1367
ZVC:Z<m>:I<n>:BANDwidth?....1367
ZVC:Z<m>:I<n>:DATA:HEADer?....1378
ZVC:Z<m>:I<n>:DATA[:VALues]?....1378
ZVC:Z<m>:I<n>:IMPedance....1368
ZVC:Z<m>:I<n>:OFFSet....1368
ZVC:Z<m>:I<n>:OVERload:RSTO....1369
ZVC:Z<m>:I<n>:OVERload:VALue?....1369
ZVC:Z<m>:I<n>:POSition....1369
ZVC:Z<m>:I<n>:SCALe....1370
ZVC:Z<m>:I<n>:SHUNt:EVALue....1370
ZVC:Z<m>:I<n>:SHUNt:MAXCurrent....1371
ZVC:Z<m>:I<n>:SHUNt:MAXVoltage....1372
ZVC:Z<m>:I<n>:SHUNt:MODE....1370
ZVC:Z<m>:I<n>:SHUNt:MXCValue?....1371
ZVC:Z<m>:I<n>:SKEW....1372
ZVC:Z<m>:I<n>:ZERComp:DETect....1376
ZVC:Z<m>:I<n>:ZERComp:STATe?....1377
ZVC:Z<m>:I<n>:ZERComp:USE....1377
ZVC:Z<m>:I<n>[:STATe]....1372

ZVC:Z<m>:ID:NAME?.... 1375
ZVC:Z<m>:ID:PARTnumber?.... 1376
ZVC:Z<m>:ID:SRNumber?.... 1376
ZVC:Z<m>:ID:SWVersion?.... 1376
ZVC:Z<m>:V<n>:BANDwidth?.... 1373
ZVC:Z<m>:V<n>:DATA:HEADer?.... 1378
ZVC:Z<m>:V<n>:DATA[:VALues]?.... 1378
ZVC:Z<m>:V<n>:IMPedance.... 1373
ZVC:Z<m>:V<n>:OFFSet.... 1373
ZVC:Z<m>:V<n>:OVERload:RSTO.... 1374
ZVC:Z<m>:V<n>:OVERload:VALue?.... 1374
ZVC:Z<m>:V<n>:POSition.... 1374
ZVC:Z<m>:V<n>:SCALe.... 1374
ZVC:Z<m>:V<n>:SKEW.... 1375
ZVC:Z<m>:V<n>[:STATe].... 1375

Index

Symbols

8b/10b
Trigger settings ... 927
Triggering ... 930
100BASE-T1
Configuration settings ... 777
Configuring ... 781
Trigger settings ... 782
Triggering ... 787
1000BASE-T1
Configuration settings ... 795
Configuring ... 798
Trigger settings ... 798
Triggering ... 804

A

Acquisition
Decimation ... 129
Modes ... 113
Single, multiple ... 36
Start and stop ... 35
Time ... 114
Acquisition modes ... 113
Acquisition time ... 114
Actions
Mask test ... 405
Active probe
Differential ... 118
Micro button ... 148
Offset compensation ... 164
Overview ... 117
ProbeMeter ... 149
Setup ... 152
Active waveform ... 49
Adapter for Tektronix probes ... 169
ADC overflow ... 1250
ADC sample rate ... 111
ADC samples ... 111
Aligning
Input channels ... 102
Alignment ... 100
Amplitude measurement ... 317
Amplitude/time measurements
Settings ... 321
Analog
Parallel bus display ... 989
Annotate ... 57
App Cockpit key ... 41
Appearance
Color tables ... 76
Application, external ... 107
Area level
Amplitude/time measurements ... 322
Area measurements ... 320
Arithmetic ... 130
Arranging waveforms ... 50
Audio
Configuring ... 536
Track ... 542
Trend ... 545
Trigger settings ... 537
Auto clear
Search results ... 439
Auto Norm
Key (trigger) ... 35
Auto trigger mode ... 35
Auto-logon ... 1224
Auto, trigger mode ... 220
Autocorrelation ... 265
Automatic measurements
Configuration settings ... 299
Display settings ... 303
Gate settings ... 305
Autonaming ... 104
Autoset
for protocols ... 487
Key ... 40
AutoZero ... 147
Average ... 130
FFT ... 383
Mathematics ... 272
Average count ... 127, 131, 273
Average count (N-single count) ... 127, 131, 273

B

Bandwidth ... 141
Instrument ... 111
Probe ... 111
Spectrum measurement ... 335
Bench top operation ... 24
Biased correlation ... 265
Big endian
CAN ... 591
LIN ... 613
Bit pattern editor ... 493
Bit string
serial pattern, MSO ... 1007
Blackman Harris window (FFT) ... 381
Brochure ... 20
Bug report ... 1256
Burst width ... 318
Bus key ... 38

C

C<n> keys ... 37
Camera key ... 40
CAN
Configuration settings ... 579
Data length ... 591
Data pattern ... 591
Endianness ... 591
Frame types ... 589
ID type ... 589
Identifier ... 590
Transfer order ... 591
Trigger settings ... 586
Trigger types ... 587
Center frequency
Spectrum measurement ... 337
Channel
Bandwidth measurement ... 337
C<n> keys ... 37

Center frequency ... 337
Connector ... 30
Offset ... 139
Waveforms ... 48
Channel power
Spectrum measurement ... 335
Clear status
Remote ... 1288
Clipping ... 1250
Clock
Parallel bus ... 989
Clock period
SENT configuration ... 640
Clock tolerance
SENT configuration ... 640
Color tables ... 76
Comb
Parallel bus display ... 989
Command sequence
Remote ... 1292
Commands
Finding a command ... 1258
Compensation, passive probes ... 185
Computer name
Changing ... 1230
Configuration
hardware ... 103
SENT ... 639
Connectors
Probe compensation ... 31
Rear panel ... 31
USB ... 31
Coupled zoom ... 248
quick access ... 56, 241
Coupling ... 109
Channel ... 140
Trigger ... 237
Waveform generator settings ... 1038
CRC calculation
SENT configuration ... 641
CRC version
SENT configuration ... 641
Cross-correlation ... 265
CSV
Waveform export ... 457
CSV export ... 107
Cursor
Saving data ... 473
Settings for data export ... 460
Used for gating ... 305
Cursor key ... 40
Cursor measurements
Disabling ... 286
Enable ... 288
Envelope waveform ... 291
How to ... 285
Peak excursion ... 295, 338
Peak search ... 293
Results ... 284
Settings ... 288
Cursors ... 56
Configuring display ... 288
Configuring measurement ... 286
Coupling ... 292
Display ... 292
Position ... 290
Start measurement ... 286
Track waveform ... 286, 291
Type ... 290
Custom:
Filter settings ... 572
Custom: Manchester / NRZ
Configuration settings ... 549
Configuring ... 566
Decoding ... 575
Trigger settings ... 567
Triggering ... 572
Customer support ... 1255
CXPI
Configuration settings ... 667
Configuring ... 670
Trigger settings ... 671
Triggering ... 678
Cycle area measurement ... 320
Cycle measurements ... 317

D

D-PHY
Basics ... 875
Configuration settings ... 875
Configuring ... 879
Triggering ... 885
Data
SENT configuration ... 640
Data entry ... 61
Data length
LIN ... 613
Data Length Code
CAN ... 591
Data nibbles
SENT configuration ... 640, 663
Data pattern
CAN ... 591
LIN ... 613
Data security ... 1256
Data sheet ... 20
DDR
Configuration settings ... 964
Configuring ... 967
Decode results ... 971
Decimation ... 113, 129
Decode
Custom protocol, decode layer ... 565
CXPI, decode layer ... 670
D-PHY, decode layer ... 878
DDR, decode layer ... 966
Ethernet, decode layer ... 729, 780, 796
M-PHY, decode layer ... 895
MDIO, decode layer ... 745
SENT ... 659
SpaceWire, decode layer ... 718
USBPD, decode layer ... 865
Decode layer
Custom protocol ... 565
CXPI ... 670
D-PHY ... 878
DDR ... 966
Ethernet ... 729, 780, 796
M-PHY ... 895
MDIO ... 745
SpaceWire ... 718
USBPD ... 865

Decode table
MSO ... 995
Decoded bus
Parallel bus display ... 989
Default
File names ... 106
Path for saving ... 106
Default values
Remote ... 1290
Delay ... 318
Delay measurement ... 323
Delete ... 58
Demo ... 56
Device footprint ... 1256
DHCP ... 1229
DHCP server
LAN configuration ... 1230
Diagrams ... 43
Dialog boxes
Usage ... 60
Dialogs ... 46
Differential probe ... 118
Digital channels ... 985
Configuration settings ... 985
Configuring ... 993
Decode table ... 995
Export ... 1010
Search ... 1011
Triggering ... 1009
Digital filter ... 143
Digital resolution ... 991
Digital signals ... 985
Digital waveforms ... 48
Display
Intensity ... 92
Overview ... 42
XY-diagram ... 280, 282
Zoom ... 241
Display elements ... 46
Diagram ... 42
Signal bar ... 42
DisplayPort ... 28, 33
DNS server
LAN configuration ... 1230
Duty cycle
Eye ... 332
Duty cycle measurement ... 318

E

Edge
Search conditions ... 423
Trigger ... 196
Edge count ... 321
Edge trigger
MSO ... 998
Electric power (math. function) ... 266
Enable registers
Remote ... 1290
Endianness
CAN ... 591
LIN ... 613
Envelope ... 130
FFT ... 383
Mathematics ... 272
Measurements ... 300
Envelope waveform
Cursor measurement ... 291
Error report ... 1256
ESR ... 1246
Ethernet
Configuration settings ... 727
Configuring ... 731
Trigger settings ... 732
Triggering ... 736
Event actions
Mask test ... 405
Event status enable register (ESE)
Remote ... 1288
Event status register (ESR)
Remote ... 1288
Export
Data ... 460
Digital channels ... 1010
History ... 473
Include horizontal values ... 462
Integer format ... 461
Parallel buses ... 1010
Raw values ... 461
Waveform histograms ... 465
XML, CSV, BIN ... 453
Expressions ... 261
External application ... 107
External monitor ... 28, 33
External trigger input ... 33, 236
External trigger out ... 225
External trigger output ... 33
Extinction ratio
Eye ... 332
Eye
Amplitude ... 332
Base ... 332
Bit rate ... 332
Fall time ... 332
Height ... 332
Jitter ... 332
Measurement results ... 332
Rise time ... 332
Top ... 332
Width ... 332
Eye measurements
Characteristics ... 330
Results ... 332
Settings ... 333

F

Fail condition, mask test ... 405
Fail criteria, mask test ... 394
Fall time ... 318
FFT ... 57, 370
Configuring ... 374
Fundamentals ... 370
Gating ... 384
Key ... 38
measurement speed ... 373
MSO ... 1011
Peak excursion ... 295, 338
Performance considerations ... 375
Search ... 419
Setup ... 378, 382
Window types ... 381

File formats
Waveform export ... 453
File import
Restrictions ... 453
File names
Default ... 104, 106
Filter
Custom ... 572
Find
Remote control command ... 64
Find trigger level ... 56
FIR (math. function) ... 266
FIR filter ... 269
Flattop2 window (FFT) ... 381
FlexRay
Configuration settings ... 618
Trigger settings ... 623
Triggering ... 628
Formula editor
Reference ... 261
Using ... 268
Formulas
Advanced expressions ... 261
Operator editor ... 259
Frame length
SENT configuration ... 641
Frame types
CAN ... 589
Frontend
information ... 103
Frontpanel
information ... 103
Function generator option ... 34

G

Gain, vertical ... 109
Gate
Coupled to cursor ... 305
Coupled to zoom ... 305
Saving data ... 473
Search ... 435
Search, defining ... 437
Settings ... 305
Settings for data export ... 460
Gate area
Configuring ... 304
Coupling to zoom ... 306, 385, 437
Gating
FFT ... 384
Gaussian window (FFT) ... 381
GBIP connector ... 34
Gen key ... 38
Getting started ... 18
Glitch
Search conditions ... 424
Trigger ... 197
GPIB
Address ... 1242
Remote control interface ... 1241
GPIB bus control
Remote ... 1289
Grid ... 44
Grounding ... 26

H

Hamming window (FFT) ... 381
Hann window (FFT) ... 381
Hardware
information ... 103
Hardware check ... 103
Hardware zoom ... 56, 241
HD key ... 40
HDMI ... 33
HDMI connector ... 33
Header files ... 453
Help ... 18, 55
Open ... 64
Search for topic ... 65
Using ... 65
High definition ... 133
High res ... 129
High signal level measurement ... 317
Histogram
Saving ... 452
Histogram measurements
Settings ... 346
Histograms ... 57, 340
Area ... 46
averaging ... 314
Characteristics ... 340
Configuring ... 344
Diagram ... 46
Exporting waveform histograms ... 465
Horizontal ... 340
Jitter measurement ... 343
Markers ... 347
Measurement histograms ... 359
Measurement results ... 340
Saving waveform histograms ... 465
Setup ... 345
Toolbar icons ... 344
Tubes ... 314
Vertical ... 340
History
Export ... 460, 473
MSO ... 994
time stamp ... 278
History key ... 40
Hold time ... 324
Holdoff ... 221
TV trigger ... 215
Using ... 192
Horizontal
Controls ... 36
Label ... 45
Position ... 36
Horizontal position ... 115
Hysteresis
Reference level ... 308
Trigger ... 222

I

I²C
Basics ... 494
Configuration settings ... 496
Configuring ... 499, 622
Decode results ... 505
Trigger settings ... 499
Triggering ... 504, 541

Icons
Measurement status ... 302
ID type
CAN ... 589
Identification
Remote ... 1288
Identifier
CAN ... 590
LIN ... 612
Import waveforms
Restrictions ... 453
Infinite persistence ... 91, 979
Input ... 30
Instrument settings
Loading ... 445
Recall ... 1290
Save ... 1290
Saving ... 450
Intensity ... 92
Intensity key ... 39
Intensity of display elements ... 39
Interfaces
GPIB ... 1242
LAN ... 1241
Interlaced scanning ... 219
Interleaved sample rate ... 112
Interleaved X/Y ... 462
Interpolation ... 128
Interval
Search conditions ... 428
Trigger ... 204
IP address ... 1241
Changing ... 1229
Remote Desktop ... 1239
IST ... 1246
IST flag
Remote ... 1289

J

Jitter
Measurement via histogram ... 343
Jitter measurements
Settings ... 1071

K

Kaiser Bessel window (FFT) ... 381
Keep X-grid fixed ... 79
Keep Y-grid fixed ... 79
Keyboard
On-screen ... 61
Keyboard, connecting ... 28
Keypad ... 61

L

Label list
SENT ... 649
Labels
Waveform ... 83
LAN ... 1233
Configuration ... 1228, 1233
Connector ... 33
Environment ... 1228
Interface ... 1241
IP address ... 1241
LAN configuration ... 1234
Ping ... 1236
Remote control interface ... 1241
Resource string ... 1241
Level knob ... 35
Level measurements ... 317
Limit checks
Actions ... 366
Configuring ... 364
Enable ... 365
Performing ... 364
Violation settings ... 366
Limits
Range ... 365
LIN
Configuration settings ... 606
Configuring ... 610
Data length ... 613
Data pattern ... 613
Endianness ... 613
Identifier ... 612
Transfer order ... 613
Trigger settings ... 610
Trigger types ... 611
Triggering ... 614
Little endian
CAN ... 591
LIN ... 613
Load instrument settings ... 1290
Loading
User settings ... 445
Lock touchscreen ... 40
Logic analyzer ... 984
Logic key ... 38
Logic probe connector ... 34
Logical operations on channels ... 211, 227
Logical thresholds
MSO, configuring ... 994
MSO, settings ... 986
Login ... 1224
Long-term measurements
Basics ... 357
Configuring ... 357
Horizontal scale ... 362
Number of points ... 362
Period of time per point ... 362
Saving ... 469
Scale mode ... 362
Settings ... 359
Low signal level measurement ... 317

M

M-PHY
Basics ... 892
Configuration settings ... 893
Configuring ... 895
Decode results ... 911
Trigger settings ... 896
Mainboard
information ... 103
Margins
Range ... 365
Marker limit
Histograms ... 347
Markers
Histograms ... 340, 347

Mask definition
Settings ... 398
Type ... 397
Mask setup ... 398
Mask test ... 57, 394
Appearance settings ... 408
Display settings ... 407, 408
Display, configuring ... 416
Event actions, configuring ... 416
Event actions, settings ... 405
Fail condition ... 405
Fail criteria ... 394
Fail criteria, configuring ... 416
History, testing ... 418
Performing ... 417
Results ... 394
Setting up ... 416
Starting ... 417
State, mask test result ... 417
Test settings ... 396
Violation tolerance ... 405
Mask test results
Acq. completed ... 395
Acq. remaining ... 395
Acquisition hits ... 395
Fail rate ... 395
Sample hits ... 395
State ... 395
Test result ... 396
Mask testing ... 394
Masks
Creating user mask ... 411
Definition ... 394, 398
Load settings ... 397
Loading ... 417
modifying user mask ... 412
Save settings ... 397
Saving ... 417
Segments, point settings ... 400
Segments, settings ... 399
User mask definition ... 398
Waveform mask definition ... 401
Math key ... 38
Math waveforms ... 38, 48
Arithmetic ... 272
Displaying ... 257
FFT ... 370
FFT Gating ... 384
FFT setup ... 378, 382
Formula editor ... 261
Operator editor ... 259
Operators ... 260, 261
Saving ... 257
Scale ... 271
Setup ... 258
Max value measurement ... 317
Maximum sample rate ... 112
MDIO
Configuration settings ... 743
Configuring ... 746
Trigger settings ... 747
Triggering ... 750
MDIO protocol
Basics ... 742
Mean measurement ... 317
Measure key ... 40, 297
Measurement ... 57
Measurement histograms ... 359
Measurement types
Area ... 320
Arithmetic ... 321
Counting ... 321
Eye ... 332
Histograms ... 340
Protocol ... 350
Spectrum ... 335
Time ... 318
Measurements
Amplitude/time ... 316
Configuring ... 297
Delay ... 323
Icons in results table ... 302
Pulse train count ... 326
Results ... 301
Setup/Hold ... 324
Source for analysis ... 359
Starting ... 297
Statistics ... 356
Median
Histograms ... 340
Memory ... 274
Menu ... 46
Micro button ... 118, 148
Min value measurement ... 317
Mixed signal option ... 984
Mixed Signal Option ... 34
Monitor ... 28, 33, 93
Mouse, connecting ... 28
MSO ... 34, 984
FFT ... 1011
History ... 994
Resolution ... 991
Time qualification ... 1005
Trigger settings ... 997
Zoom ... 994
Multi channel export
Enable ... 460
Remote control ... 1342, 1649
Results in header file ... 456
Multiple measurements ... 362

N

N dB down
Spectrum ... 337
Network
Environment ... 1228
Noise
Eye ... 332
Hysteresis settings ... 222
Normal trigger mode ... 35
Normal, trigger mode ... 220
Notification ... 45
Numeric data entry ... 61

O

Occupied bandwidth
Spectrum measurement ... 335, 338
OCXO ... 125
Offset ... 38
Active probe ... 164
Channel ... 139
On/Off key ... 34

One File ... 446
Open source acknowledgment ... 20
Operating system ... 1223
Access ... 1227
service packs ... 1225
Settings ... 1227
Operation
Concepts ... 41
Manual ... 41
Operation complete
Remote ... 1289
Operator
Type ... 260
Operators
Formula editor mathematics ... 261
Optical
Average power ... 348
Modulation amplitude ... 348
Options
Identification (remote) ... 1289
K510, I²S, Audio ... 529
K520, FlexRay ... 618
R&S RTO6-B1 (MSO) ... 34
R&S RTO6-B1E (for R&S RT-ZVC) ... 34
R&S RTO6-B6 (waveform and function generator) ... 34
R&S RTO6-B10 (GBIP) ... 34
Output buffer ... 1246
Overshoot measurement ... 317

P

Parallel buses ... 985
Clock setup ... 989
Configuration settings ... 985
Configuring ... 993
Decode table ... 995
Export ... 1010
Triggering ... 1009
Parallel poll register enable
Remote ... 1290
Passive probe
Overview ... 117
Settings ... 150
Password ... 1224
Remote Desktop ... 1239
Pattern
Qualification, state trigger ... 211, 227
Pattern generator
Setup ... 1032
Pattern search condition ... 210
Pattern trigger ... 210
MSO ... 1005
Pause pulse
SENT configuration ... 641
PCIe
Trigger settings ... 942
PCIe Gen 1/2
Search settings ... 953
Searching ... 963
Triggering ... 951
Peak detect ... 129
Peak excursion
Cursor measurement ... 295, 338
Search ... 295, 338
Spectrum measurement ... 295, 338
Peak search
Cursor measurement ... 285, 286
Spectrum measurement ... 285, 293
Peak to peak measurement ... 317
Performance ... 94
Considerations for FFT ... 375
FFT parameters ... 373
Performance test ... 125
Ping ... 1236
Position ... 38
Horizontal ... 36, 115
Vertical ... 38, 109
Position mode ... 82
Power
Connector ... 32
Key ... 34
On/Off ... 26
PPE ... 1246
Preset ... 98
Preset key ... 40
Printing
Settings ... 476
Probability markers
Histograms ... 347
ProbeMeter ... 118
Measurement ... 317
Setting ... 149
Probes
Active ... 117
AutoZero ... 147
Compensation ... 31
Differential ... 118
Passive ... 117
Passive, compensation ... 185
Setup ... 145
Tektronix ... 169
Progressive scanning ... 219
Projector ... 28
Protocol
SENT ... 635
Protocols
8b/10b trigger settings ... 927
100BASE-T1 configuration settings ... 777
100BASE-T1 trigger settings ... 782
1000BASE-T1 configuration settings ... 795
1000BASE-T1 trigger settings ... 798
Audio trigger settings ... 537
CAN configuration settings ... 579
Custom: filter settings ... 572
Custom: Manchester / NRZ configuration settings ... 549
Custom: Manchester / NRZ trigger settings ... 567
CXPI configuration settings ... 667
CXPI trigger settings ... 671
D-PHY configuration settings ... 875
Data format ... 487
DDR configuration settings ... 964
DDR decode results ... 971
Display settings ... 487
Ethernet configuration settings ... 727
Ethernet trigger settings ... 732
FlexRay configuration settings ... 618
FlexRay trigger settings ... 623
I²C configuration settings ... 496
I²C decode results ... 505
I²C trigger settings ... 499
LIN configuration settings ... 606
LIN trigger settings ... 610

M-PHY configuration settings ... 893
M-PHY decode results ... 911
M-PHY trigger settings ... 896
MDIO configuration settings ... 743
MDIO trigger settings ... 747
PCIe Gen 1/2 search settings ... 953
PCIe trigger settings ... 942
SENT ... 634
SENT configuration ... 639
SENT decode results ... 659
SENT label list ... 649
SENT protocol ... 635
SENT search ... 660
SENT trigger ... 643
SpaceWire configuration settings ... 715
SPI configuration settings ... 510
SPI decode results ... 517
SPI trigger settings ... 515
UART decode results ... 527
USB 2.0 configuration settings ... 818
USB 2.0 trigger settings ... 824
USB 3.1 configuration settings ... 849
USB 3.1 trigger settings ... 853
USBPD configuration settings ... 863
USBPD trigger settings ... 866
Pulse count ... 321
Pulse train count ... 326

Q

Q-factor
Eye ... 332
Quick measurements ... 57

R

R&S RT-ZVC
multi-channel power probe ... 171
R&S RT-Z2T ... 169
Rack mounting ... 25
Recall instrument settings ... 1290
Recall intermediate ... 1290
Record length ... 111, 114
Rectangular window (FFT) ... 381
Redo ... 55
Ref key ... 37
Reference levels
Basics ... 307
Configuring ... 308
Defining automatically ... 308
Defining manually ... 308
displaying ... 304
Displaying ... 310
Hysteresis ... 308
Level settings ... 310
Settings ... 310
Tubes ... 308
Reference point ... 36, 45, 115
Reference signal
1 GHz, internal ... 125
External ... 125
OCXO ... 125
Reference waveforms ... 37, 48
Displaying ... 249
Loading ... 249
Saving ... 249
Registers ... 1246
Release notes ... 20
Remote commands
Finding a command ... 1258
Remote control ... 41
Find command using help ... 64
GPIB address ... 1242
Interfaces ... 1241
Protocols ... 1241
Starting ... 1244
Remote Desktop ... 41, 1238
Remote operation ... 41, 1236, 1238
Repetitive, trigger mode ... 220
Reset
Histogram, long-term meas., statistics ... 361
Reset values
Remote ... 1290
Resolution ... 109, 111, 114
16 bit ... 133
Resolution enhancement ... 113
Resource string
LAN ... 1241
Restoring
Settings ... 99
Result boxes
Displaying ... 58
Result lines
Displaying ... 304
Result tables ... 46
Results
Clear ... 303
Configuring display ... 58
Lines in display ... 310
Mask test ... 394
Measurements ... 301
Reset long-term measurement ... 361
Saving ... 467
RFFE
Advanced ... 759
Trigger settings ... 760
Triggering ... 768
Rise time ... 318
RMS
Acquisition setting ... 129
RMS measurement ... 317
Roll mode ... 121
Strart roll time ... 122
RS232 ... 520
Basics ... 520
RTO6-B10 ... 34
RTxServiceReporter ... 1256
Run / Stop ... 55
Run Stop key ... 35
Runt
Search conditions ... 425
Trigger ... 199

S

Safety instructions ... 19
Sample rate ... 111
Samples ... 111
History ... 274
Save instrument settings ... 1290
Save intermediate ... 1290
Save Reference ... 58
Save Waveform ... 58

Save/Recall
Remote ... 1640
User preferences ... 448
User settings ... 445, 446
Waveforms ... 460
Saveset ... 445, 450
Saving
File name generation ... 104
History ... 460, 473
Instrument settings ... 445, 450
Long-term measurements ... 469
Preset ... 98
Results ... 467
Screenshots ... 479
User preferences ... 448
Waveform histograms ... 465
Waveform segments ... 460, 473
Waveforms ... 460, 473
Scale
Horizontal, rotary knob ... 36
Math waveforms ... 271
Vertical ... 109, 139
Vertical, rotary knob ... 39
Scaling
Horizontal ... 120
Long-term measurements ... 361
Long-term measurements, horizontal ... 362
Measurement histograms ... 361
Scanning system (TV signals) ... 219
SCPI
Finding a command ... 1258
Screen resolution ... 28
Screenshot
Saving ... 479
Web interface ... 1237
Screenshots ... 476
Meta information ... 475
Print and save ... 475
Search ... 56
Auto clear ... 439
Clearing results ... 439
Conditions ... 419
Control ... 419
Copy trigger settings ... 422
Definition ... 419
Edge ... 423
Enable ... 421
FFT ... 419
Gate ... 435
Glitch ... 424
Interval ... 428
On digital signals ... 1011
PCIe Gen 1/2 settings ... 953
Peak excursion ... 295, 338
Result box ... 420
Runt ... 425
Scope ... 419
SENT ... 660
SENT error condition ... 664
SENT search criteria ... 661
SENT settings ... 660
SENT short serial message ... 663
SENT transmission sequence ... 662
Setup & Hold ... 430
Show results ... 439
Slew rate ... 429
Sorting results ... 439
Source ... 421
Timeout ... 427
Trigger level ... 423, 424, 425, 428, 429, 431
Trigger search ... 422
Width ... 424
Window ... 426
Zoom window, configuring ... 442
Search conditions
Pattern ... 210
Searching
In help ... 65
Secured environment ... 1256
Select
Waveform ... 50
Selected waveform ... 49
Self-alignment ... 100, 102, 1288
Self-test
Remote ... 1291
Selftest ... 103
SENT
About ... 634
Configuration ... 639
Configuring ... 642
Decode results ... 659
Label list ... 649
Protocol ... 635
Search ... 660
Search criteria ... 661
Search error condition ... 664
Search settings ... 660
Search short serial message ... 663
Search transmission sequence ... 662
Trigger error condition ... 648
Trigger serial message setup ... 646
Trigger settings ... 643
Trigger transmission sequence ... 645
Trigger types ... 643
Triggering ... 649
SENT configuration
Clock period ... 640
Clock tolerance ... 640
CRC calculation ... 641
CRC version ... 641
Data ... 640
Data nibbles ... 640, 663
Frame length ... 641
Pause pulse ... 641
Serial protocol ... 640
Threshold ... 640
Sequence
Search ... 663
Trigger ... 646
Serial message
Search sequence ... 663
Sequence ... 646, 663
Trigger sequence ... 646
Serial pattern trigger
MSO ... 1007
Serial Pattern trigger ... 213
Serial protocol
SENT configuration ... 640
Service manual ... 19
Service packs ... 1225
Service request enable register (SRE)
Remote ... 1291
Settling time ... 318

Setup & Hold
Search ... 430
Trigger ... 207
Trigger (MSO) ... 1002
Setup time ... 324
Setup/Hold measurement ... 324
Setup/Hold ratio ... 324
Show labels ... 82
Signal bar ... 45
Signal icons ... 45
Signal label ... 45
Signal levels
Configuring ... 308
Signal preview ... 49
Signal threshold
Amplitude/time measurements ... 322
Signal view ... 45
Signals
Color settings ... 75
Single ... 55
Single key ... 36
Skew ... 123
Slew rate
Trigger ... 205
Slew Rate
Search conditions ... 429
Slope key ... 35
SmartGrid ... 50
Source key ... 35
SpaceWire
Advanced ... 717
Basics ... 714
Configuration settings ... 715
Configuring ... 718
Spectrogram ... 58
Spectrum measurements
Channel BW ... 337
Channel CF ... 337
Cursor measurement ... 285, 286
N dB down ... 337
Occupied bandwidth ... 338
Peak detection ... 286
Peak excursion ... 295, 338
Peak search ... 285
Results ... 335
Settings ... 336
SPI
Configuration settings ... 510
Configuring ... 514
Decode results ... 517
Trigger settings ... 515
Triggering ... 517
SPI protocol
Basics ... 509
SRE ... 1246
Standard
TV trigger ... 216
Standard deviation
Histograms ... 340
Standard deviation measurement ... 317
State trigger ... 209
MSO ... 1004
Statistics
Enable ... 359
Measurement results ... 356
Status
Measurements ... 302
Status byte
Remote ... 1288, 1291
Status registers
Overview ... 1245
STATus:OPERation ... 1246
Status reporting system
Common commands ... 1287
Status reports ... 1245
STB ... 1246
Style (waveforms) ... 92
Support ... 1256
Switch off
Waveform ... 50
Switch on
Waveform ... 49
Switching of transitions ... 321

T

Technical support ... 1256
Temperature
Changes ... 102
Termination ... 109
Text entry ... 61
Threshold
SENT configuration ... 640
Thresholds
MSO, configuring ... 994
MSO, settings ... 986
Time base ... 36, 114
Time measurements ... 318
Time qualification
MSO ... 1005
Time scale ... 36, 114, 120
Time stamp ... 278
Timebase ... 120
Timeout
Search conditions ... 427
Trigger ... 203
Timeout trigger
MSO ... 1001
Toolbar ... 52
Annotate ... 57
Coupled zoom ... 56, 241
Cursor ... 56
Delete ... 58
FFT ... 57
Find level ... 56
Hardware zoom ... 56, 241
Hide/show icons ... 53
Histogram ... 57
Histogram icons ... 344
Load saveset ... 55
Masks ... 57
Measure ... 57
Measurement ... 297
Overview ... 53
Quick measurement ... 57
Redo ... 55
Run / Stop ... 55
Save Reference ... 58
Save Waveform ... 58
Search ... 56
Show tooltips ... 55
Single ... 55
Undo ... 54

Zone trigger ... 58
Zoom ... 56, 241
Tooltips ... 55
Show ... 64
Total harmonic distortion
Spectrum measurement ... 335
Touch Lock key ... 40
Touchscreen
Adjusting ... 97
Compared with mouse ... 41
Control elements ... 46
Lock, unlock ... 40
Usage ... 41
Track ... 542
Enable ... 359
Track waveform
Cursor measurement ... 286, 291
Transfer order
CAN ... 591
LIN ... 613
Transition switching count ... 321
Transition trigger
See Slew rate trigger ... 205
Transparency ... 39
Trend ... 359, 545
Trigger
8b/10b settings ... 927
100BASE-T1 settings ... 782
1000BASE-T1 settings ... 798
Audio settings ... 537
CAN settings ... 586
Controls ... 35
Custom: Manchester / NRZ settings ... 567
CXPI settings ... 671
Delay (Holdoff) ... 221
Edge ... 196
Edge (MSO) ... 998
Ethernet settings ... 732
Event (definition) ... 190
Event (remote) ... 1291
External input ... 33, 236
External output ... 33
FlexRay settings ... 623
Force ... 221
Glitch ... 197
Holdoff ... 221
I²C settings ... 499
Information ... 191
Interval ... 204
Label ... 45
Level ... 35, 44
Level, search ... 423, 424, 425, 428, 429, 431
LIN settings ... 610
M-PHY settings ... 896
MDIO settings ... 747
Mode ... 35, 220
MSO settings ... 997
Pattern ... 210
Pattern (MSO) ... 1005
PCIe settings ... 942
Position ... 44
Run ... 220
Runt ... 199
Search ... 422
SENT ... 643
SENT error condition ... 648
SENT serial message setup ... 646
SENT transmission sequence ... 645
SENT trigger type ... 643
Sequence (definition) ... 190
Serial Pattern ... 213
Serial pattern (MSO) ... 1007
Settings, copy to search ... 422
Setup & Hold ... 207
Setup & Hold (MSO) ... 1002
Slew rate ... 205
Slope ... 35
Source ... 35
Sources, MSO ... 997
SPI settings ... 515
State ... 191, 209
State (MSO) ... 1004
Timeout ... 203
Timeout (MSO) ... 1001
TV (video) ... 215
USB 2.0 settings ... 824
USB 3.1 settings ... 853
USBPD settings ... 866
Width ... 198
Width (MSO) ... 999
Window ... 201
Trigger level
Find ... 56
Trigger offset ... 115
Trigger out
Pulse on limit violation ... 367
Settings ... 225
Trigger sources
MSO ... 997
Trigger types
CAN ... 587
Edge ... 196
Glitch ... 197
Interval ... 204
LIN ... 611
Pattern ... 210
Runt ... 199
Setup & Hold ... 207
Slew rate ... 205
Timeout ... 203
Width ... 198
Window ... 201
Trigger Types ... 195
NFC ... 219
Serial Pattern ... 213
State ... 209
TV/Video ... 215
Tubes
Configuring ... 314
Histograms ... 314
Reference level ... 308
TV trigger ... 215
Type
Trigger ... 195

U

UART ... 520
Basics ... 520
Configuring ... 524
Decode results ... 527
Triggering ... 527
Unbiased correlation ... 265
Undersampling ... 111

Undo 54
Unlock touchscreen 40
USB
Configuring 823
Connecting devices 28
Connector 31
Device 33
Triggering 856
USB 2.0
Configuration settings 818
Trigger settings 824
Triggering 831
USB 3.1
Configuration settings 849
Configuring 852
Trigger settings 853
USB 3.1 protocol
Basics 840
USB protocol
Basics 813
USBPD
Configuration settings 863
Configuring 866
Trigger settings 866
Triggering 869
User ID
Remote Desktop 1239
User manual 19
User mask
Settings 398
User name 1224
User settings
Saving 445

V

Vertical
Controls 37
Position 109
Resolution 109
Scale 109, 139
Vertical system 109
Video (TV) trigger 215
Violation
Limit checks 366
Violation of limits
Actions 366
Violation tolerance, mask test 405
Virus protection 1225
VISA
Resource string 1241
VNC 41, 1236

W

Wait
Remote 1292
Waveform
Arithmetic 113
Data query 1342
Loading 452
Saving 452
Waveform arithmetic 130
Waveform count
Histograms 340
Waveform export
CSV 457
Data query 1342
Files and formats 453
Header files 453
Remote control 1649
Value files 457
XML 457
Waveform generator
AM modulation 1022
Arbitrary 1027
Coupling 1038
DC offset alignment 1037
FM modulation 1024
FSK modulation 1026
Function generator 1015
Modulation 1020
Noise 1036
Output 1035
PWM modulation 1023
Settings 1013
Sweep 1026
Waveform generator option 34
Waveform mask
Settings 401
Waveform samples 111
Waveform Value files 457
Waveforms 48
Arrange 50
Clipping 1250
Overview and usage 48
Preview 50
Saving 460, 473
Select 50
States 49
Style 92
Switch off 50
Switch on 49
Zooming 239
Web browser 1233
Web control 1236
Web interface 41, 1233
Browser 1233
LAN configuration 1234
Width
Search conditions 424
Trigger 198
Width trigger
MSO 999
Window
Search conditions 426
Trigger 201
Windows 1223
Access 1227
Settings 1227

X

XML
Waveform export 457
xx
Advanced 779
XY-diagram
Displaying 282
settings 280
XY-waveforms 48

Z

Zone key 36
Zone trigger 36, 58
Zoom 239
Area 45
Area, coupling 306, 385, 437
Coupled 248
Diagram 45
Key 37
Methods 239
MSO 994
Multiple 247
On the touchscreen 245
Position/Range 247
Procedures 245
Quick access 56, 241
Saving data 473
Settings 241
Settings for data export 460
Start-stop values 246
Start/Stop settings 242
Used for gating 305
Waveforms 48

dataTec
Mess- und Prüftechnik. Die Experten.

Ihr Ansprechpartner / Your Partner:

dataTec AG
E-Mail: info@datatec.eu
>>> www.datatec.eu