

R&S®NRPxxP

Pulse Power Sensors

User Manual



1179576002
Version 04

ROHDE & SCHWARZ
Make ideas real



This manual describes the following pulse power sensors with firmware version FW 01.10 and later:

- R&S®NRP18P (1444.1190K02)
- R&S®NRP40P (1444.1290K02)
- R&S®NRP50P (1444.1390K02)

© 2023 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.

All other trademarks are the properties of their respective owners.

1179.5760.02 | Version 04 | R&S®NRPxxP

Throughout this manual, products from Rohde & Schwarz are indicated without the ® symbol, for example R&S®NRP18P is abbreviated as R&S NRP18P.

Contents

1	Safety and regulatory information.....	7
1.1	Safety instructions.....	7
1.2	Labels on the product.....	8
1.3	Warning messages in the documentation.....	9
2	Welcome.....	10
2.1	Documentation overview.....	10
2.1.1	Getting started manual.....	10
2.1.2	User manuals.....	10
2.1.3	Tutorials.....	10
2.1.4	Instrument security procedures.....	10
2.1.5	Basic safety instructions.....	10
2.1.6	Data sheets and brochures.....	11
2.1.7	Release notes and open source acknowledgment (OSA).....	11
2.1.8	Application notes, application cards, white papers, etc.....	11
2.2	Key features.....	11
3	Preparing for use.....	12
3.1	Unpacking and checking.....	12
3.2	Choosing the operating site.....	12
3.3	Considerations for test setup.....	13
3.4	Connecting to a DUT.....	13
3.5	Powering the power sensor.....	14
3.6	Connecting a cable to the host interface.....	14
3.7	Connecting to a controlling host.....	15
3.7.1	Computer.....	15
3.7.2	Android smartphone/tablet.....	18
3.7.3	Base unit.....	19
4	Power sensor tour.....	20
4.1	RF connector.....	20
4.2	Status information.....	21
4.3	Host interface.....	21

4.4	Trigger I/O connector.....	21
5	Operating concepts.....	22
5.1	R&S NRP Toolkit.....	22
5.1.1	Versions and downloads.....	22
5.1.2	System requirements.....	22
5.1.3	R&S NRP Toolkit for Windows.....	23
5.2	Remote control.....	24
5.3	R&S NRPV.....	25
5.4	R&S Power Viewer.....	26
5.5	R&S NRX.....	27
6	Firmware update.....	29
6.1	Preparing the update.....	29
6.2	Updating the firmware.....	29
6.2.1	Using the Firmware Update for NRP Family program.....	29
6.2.2	Using remote control.....	31
7	Replacing an R&S NRP-Zxx with an R&S NRPxxP.....	33
7.1	Important difference.....	33
7.2	Prerequisites.....	33
8	Remote control commands.....	34
8.1	Conventions used in SCPI command descriptions.....	34
8.2	Notations.....	35
8.3	Common commands.....	36
8.4	Preparing for the measurement.....	40
8.4.1	Selecting a measurement mode.....	40
8.4.2	Configuring the measured values.....	41
8.5	Controlling the measurement.....	42
8.5.1	Starting and ending a measurement.....	42
8.5.2	Triggering.....	43
8.5.3	Controlling the measurement results.....	46
8.5.4	Interplay of the controlling mechanisms.....	47
8.5.5	Configuring the trigger.....	51
8.6	Configuring the measurement modes.....	56

8.6.1	Continuous average measurement.....	56
8.6.2	Burst average measurement.....	59
8.6.3	Timeslot measurement.....	60
8.6.4	Trace measurement.....	62
8.6.5	Pulse analysis measurement.....	67
8.6.6	Statistics measurement.....	70
8.7	Configuring basic measurement parameters.....	74
8.7.1	Configuring auto averaging.....	74
8.7.2	Setting the frequency.....	78
8.7.3	Excluding intervals.....	78
8.7.4	Configuring corrections.....	80
8.8	Configuring measurement results.....	94
8.8.1	Setting the power unit.....	94
8.8.2	Setting the result format.....	95
8.9	Querying measurement results.....	96
8.9.1	Continuous average measurement results.....	96
8.9.2	Burst average measurement results.....	97
8.9.3	Timeslot measurement results.....	97
8.9.4	Trace measurement results.....	97
8.9.5	Pulse analysis measurement results.....	100
8.9.6	Statistics measurement results.....	107
8.10	Calibrating, zeroing.....	108
8.11	Testing.....	110
8.12	Configuring the system.....	111
8.12.1	Configuring general functions.....	111
8.13	Using the status register.....	118
8.13.1	General status register commands.....	119
8.13.2	Reading the CONDition part.....	119
8.13.3	Reading the EVENT part.....	119
8.13.4	Controlling the ENABLE part.....	120
8.13.5	Controlling the negative transition part.....	120
8.13.6	Controlling the positive transition part.....	120
9	Performing measurement tasks - programming examples.....	121

9.1	Performing the simplest measurement.....	121
9.2	Performing a buffered continuous average measurement.....	121
9.3	Performing trace measurements.....	123
9.4	Trace measurement with synchronization to measurement complete.....	124
10	Remote control basics.....	126
10.1	Remote control interfaces and protocols.....	126
10.1.1	USB interface.....	126
10.2	Status reporting system.....	127
10.2.1	Overview.....	127
10.2.2	Device status register.....	129
10.2.3	Questionable status register.....	131
10.2.4	Standard event status and enable register (ESR, ESE).....	133
10.2.5	Operation status register.....	134
11	Troubleshooting.....	140
11.1	Displaying status information.....	140
11.2	Performing a selftest.....	140
11.3	Problems during a firmware update.....	140
11.4	Contacting customer support.....	141
12	Transporting.....	142
13	Maintenance, storage and disposal.....	143
13.1	Regular checks.....	143
13.2	Cleaning.....	144
13.3	Storage.....	144
13.4	Disposal.....	144
	List of commands.....	146
	Index.....	151

1 Safety and regulatory information

The product documentation helps you use the product safely and efficiently. Follow the instructions provided here and in the following chapters.

Intended use

The power sensors are intended for accurate and uncomplicated power measurements in production, R&D and calibration labs as well as for installation and maintenance tasks. The supported base units are listed in the data sheet. Observe the operating conditions and performance limits stated in the data sheet.

Target audience

The target audience is developers and technicians. The required skills and experience in power measurements depend on the used operating concept.

The pulse power sensors are ideal for radar applications, but also suitable for universal use.

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


Operating the product

Only use the product indoors. The product casing is not waterproof.

Observe the ambient conditions such as altitude, operating temperature and climatic loads; see the data sheet.

Meaning of safety labels

Safety labels on the product warn against potential hazards.

	<p>Potential hazard Read the product documentation to avoid personal injury or product damage.</p>
---	--

1.2 Labels on the product

Labels on the product inform about:

- Personal safety
See "[Meaning of safety labels](#)" on page 8.
- Environment safety
See [Table 1-1](#).
- Identification of the product
The rear label of the power sensor contains the serial number that uniquely identifies the power sensor.
- Sensitive components
[Table 1-2](#)

Table 1-1: Labels regarding environment safety



	<p>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 144.</p>
---	---

Table 1-2: Labels regarding sensitive components

	<p>Electrostatically sensitive components Indicates sensitivity to touch. Follow the instructions in the product documentation to avoid product damage. See "Preventing electrostatic discharge (ESD)" on page 13.</p>
---	--

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.

NOTICE

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

2 Welcome

This chapter provides an overview of the user documentation and an introduction to the R&S NRPxxP.

2.1 Documentation overview

This section provides an overview of the R&S NRPxxP user documentation. Unless specified otherwise, you find the documents at:

www.rohde-schwarz.com/manual/nrpxxp

Further documents are available at:

www.rohde-schwarz.com/product/nrpxxp

2.1.1 Getting started manual

Introduces the R&S NRPxxP and describes how to set up and start working with the product. Includes basic operations and general information, e.g. safety instructions, etc. A printed version is delivered with the power sensor.

2.1.2 User manuals

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

2.1.3 Tutorials

Tutorials offer guided examples and demonstrations on operating the R&S NRPxxP. They are provided on the product page of the internet.

2.1.4 Instrument security procedures

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

2.1.5 Basic safety instructions

Contains safety instructions, operating conditions and further important information. The printed document is delivered with the instrument.

2.1.6 Data sheets and brochures

The data sheet contains the technical specifications of the R&S NRPxxP. 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/nrpxxp

2.1.7 Release notes and open source acknowledgment (OSA)

The release notes list new features, improvements and known issues of the current firmware version, and describe the firmware installation.

The "Open Source Acknowledgment" is provided on the user documentation CD-ROM, included in the delivery. It contains verbatim license texts of the used open source software.

www.rohde-schwarz.com/firmware/nrpxxp

2.1.8 Application notes, application cards, white papers, etc.

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

2.2 Key features

The pulse power sensors are members of the R&S NRP power sensors from Rohde & Schwarz.

They provide a high-speed USB interface that constitutes both the communication port and the power supply connection.

The R&S NRP power sensors are compatible with the R&S NRP-Z power sensors in both the interface (USB) and a common command subset. This compatibility makes the replacement of the old power sensors easy.

With their high sampling rate, the pulse power sensors analyze signals fast. The pulse power sensors support a wide range of measurements: average, continuously or gated peak power, trace, statistical signal analysis, automatic pulse analysis.

For a detailed specification, refer to the data sheet and the brochure.

3 Preparing for use

Here, you can find basic information about setting up the product for the first time.

• Unpacking and checking.....	12
• Choosing the operating site.....	12
• Considerations for test setup.....	13
• Connecting to a DUT.....	13
• Powering the power sensor.....	14
• Connecting a cable to the host interface.....	14
• Connecting to a controlling host.....	15

3.1 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.

3.2 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.

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.

3.3 Considerations for test setup

Give particular attention to the following aspects when handling power sensors.

EMI impact on measurement results

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 interface cables.
- Always terminate open cable ends.
- Ensure that connected external devices comply with EMC regulations.

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).

Do not touch the inner conductor of the RF connector to prevent electrostatic discharge damage.

3.4 Connecting to a DUT

For connecting the power sensor to a DUT, use the RF connector. See [Chapter 4.1, "RF connector"](#), on page 20.

To connect to the DUT

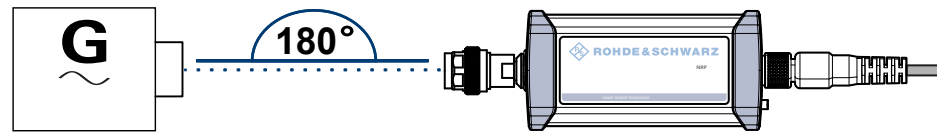
1. Ensure that the RF connector of your DUT is compatible with the RF connector of the power sensor.
See [Table 4-1](#).

2. Inspect both RF connectors carefully. Look for metal particles, contaminants and defects.

If either RF connector is damaged, do not proceed, because the risk of damaging the mating connector is too high.

See also [Chapter 13.1, "Regular checks"](#), on page 143.

3. Insert the RF connector straight into the RF output of your DUT. Take care not to tilt it.



4. **NOTICE!** Risk of damaging the center pin of the RF connector. Only rotate the hex nut of the RF connector. Never rotate the power sensor itself.
Tighten the RF connector manually.
5. Tighten the RF connector using a torque wrench with the nominal torque recommended in [Chapter 4.1, "RF connector"](#), on page 20 to ensure maximum measurement accuracy.

To disconnect from the DUT

1. **NOTICE!** Risk of damaging the center pin of the RF connector. Only rotate the hex nut of the RF connector. Never rotate the power sensor itself.
Carefully loosen the union nut at the front of the RF connector of the power sensor.
2. Remove the power sensor.

3.5 Powering the power sensor

The electrical power for the R&S NRPxxP is supplied over the host interface. See [Chapter 4.3, "Host interface"](#), on page 21.

3.6 Connecting a cable to the host interface

For connecting the power sensor to a USB host, use the host interface. See [Chapter 4.3, "Host interface"](#), on page 21.

Depending on the USB host, use one of the following cables:

- Computer or R&S NRP-Z5 sensor hub:
R&S NRP-ZKU cable with a USB connector or R&S NRP-ZKC cable with a USB-C connector
See [Chapter 3.7.1, "Computer"](#), on page 15.
- Android smartphone/tablet:
R&S NRP-ZKC cable with a USB-C connector
See [Chapter 3.7.2, "Android smartphone/tablet"](#), on page 18.
- Base units or other supported Rohde & Schwarz instruments:
R&S NRP-ZK6 cable with a push-pull type connector
See [Chapter 3.7.3, "Base unit"](#), on page 19.

These cables can be obtained in different lengths up to 5 meters. The order numbers are provided in the data sheet.



The length of the R&S NRP-ZKC cable is up to 3 meters only.

To connect a cable to the host interface of the power sensor

1. Insert the screw-lock cable connector into the host interface connector of the power sensor.
2. Tighten the union nut manually.

To disconnect the host interface of the power sensor

1. Loosen the union nut of the screw-lock cable connector.
2. Remove the cable.

3.7 Connecting to a controlling host

As a controlling host, you can use:

- [Computer](#)
- [Android smartphone/tablet](#)
- [Base unit](#)

For operating the power sensor, you can choose from various possibilities. For details, see [Chapter 5, "Operating concepts"](#), on page 22.

3.7.1 Computer

If the controlling host is a computer, you can operate the power sensor in several ways. For details, see [Chapter 5, "Operating concepts"](#), on page 22.

► Establish the connection using:

- Host interface
See [Chapter 3.7.1.1, "USB connection"](#), on page 15.
See [Chapter 3.7.1.2, "R&S NRP-Z5 sensor hub setup"](#), on page 16.

3.7.1.1 USB connection

All R&S NRPxxP power sensors can be connected to the USB interface of a computer.

Sleep mode

If a power sensor is connected to a laptop and you close the laptop, the power sensor goes automatically into sleep mode to minimize its power consumption.

If you open the laptop again, the power sensor wakes up and enumerates in the host system again. The power sensor does not recall the settings before the sleep mode and starts afresh, as after booting or rebooting.

Required equipment

- R&S NRPxxP power sensor
- R&S NRP-ZKU cable or R&S NRP-ZKC cable

Setup

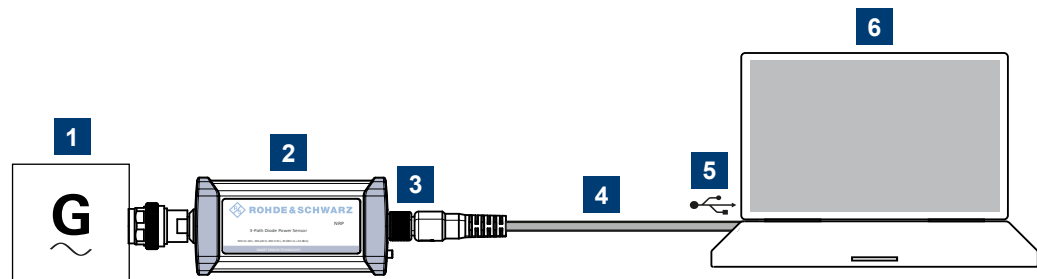


Figure 3-1: Setup with an R&S NRP-ZKU cable

- 1 = Signal source
- 2 = R&S NRPxxP power sensor
- 3 = Host interface connector
- 4 = R&S NRP-ZKU cable or R&S NRP-ZKC cable
- 5 = USB connector
- 6 = Computer with installed VISA driver or R&S NRP Toolkit

Set up as shown in [Figure 3-1](#).

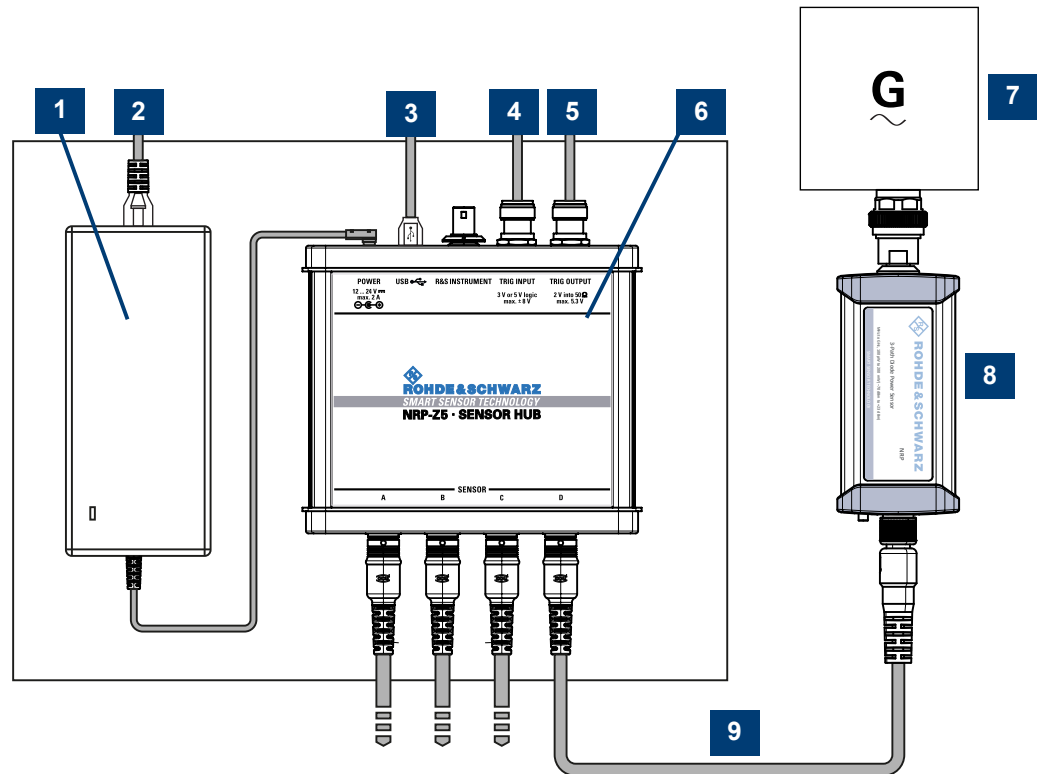
1. Connect the R&S NRP-ZKU cable or R&S NRP-ZKC cable to the power sensor. See ["To connect a cable to the host interface of the power sensor"](#) on page 15.
2. Connect the R&S NRP-ZKU cable or R&S NRP-ZKC cable to the computer.
3. **NOTICE!** Incorrectly connecting or disconnecting the power sensor can damage the power sensor or lead to erroneous results. Ensure that you connect or disconnect the power sensor as described in [Chapter 3.4, "Connecting to a DUT"](#), on page 13.
Connect the power sensor to the signal source.
4. On the computer, start a software application to view the measurement results. See [Chapter 5, "Operating concepts"](#), on page 22.

3.7.1.2 R&S NRP-Z5 sensor hub setup

The R&S NRP-Z5 sensor hub (high-speed USB 2.0) can host up to four R&S NRPxxP power sensors and provides simultaneous external triggering to all connected power sensors.

Required equipment

- 1 to 4 R&S NRPxxP power sensors
- 1 R&S NRP-ZK6 cable per power sensor
- R&S NRP-Z5 sensor hub with external power supply unit and USB cable
- BNC cables to connect the trigger input and trigger output signals (optional)

Setup**Figure 3-2: Setup with an R&S NRP-Z5 sensor hub**

- 1 = External power supply unit
- 2 = Connect to AC power supply.
- 3 = Connect to computer with USB host interface.
- 4 = Optional: Connect to trigger source.
- 5 = Optional: Connect to triggered device.
- 6 = R&S NRP-Z5 sensor hub
- 7 = Signal source (DUT)
- 8 = R&S NRPxxP power sensor
- 9 = R&S NRP-ZK6 cable

Set up as shown in [Figure 3-2](#).

1. Connect the R&S NRP-ZK6 cable to the power sensor. See "[To connect a cable to the host interface of the power sensor](#)" on page 15.
2. Connect the power sensors to the R&S NRP-Z5 sensor hub. You can connect up to four power sensors.

3. Connect the R&S NRP-Z5 to the computer.
4. **NOTICE!** Incorrectly connecting or disconnecting the power sensor can damage the power sensor or lead to erroneous results. Ensure that you connect or disconnect the power sensor as described in [Chapter 3.4, "Connecting to a DUT"](#), on page 13.
Connect the power sensors to the signal sources.
5. Connect the delivered external power supply unit to the R&S NRP-Z5 and to an AC supply connector.
6. Connect the trigger input of the R&S NRP-Z5 with a BNC cable to the trigger source (optional).
7. Connect the trigger output of the R&S NRP-Z5 with a BNC cable to the trigger device (optional).
8. On the computer, start a software application to view the measurement results. See [Chapter 5, "Operating concepts"](#), on page 22.

3.7.2 Android smartphone/tablet

If the controlling host is an android smartphone/tablet with USB-C connector, you can operate the power sensor using R&S Power Viewer Mobile. For details, see [Chapter 5, "Operating concepts"](#), on page 22.

- ▶ Establish the connection using:
 - Host interface
See [Chapter 3.7.2.1, "USB-C connection"](#), on page 18.

3.7.2.1 USB-C connection

All R&S NRPxxP power sensors can be connected to the USB-C interface of an Android smartphone/tablet.

Required equipment

- R&S NRPxxP power sensor
- R&S NRP-ZKC cable

Setup

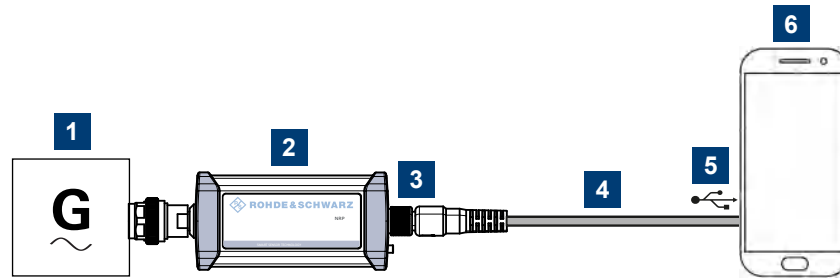


Figure 3-3: Setup with an R&S NRP-ZKC cable

- 1 = Signal source
- 2 = R&S NRPxxP power sensor
- 3 = Host interface connector
- 4 = R&S NRP-ZKC cable
- 5 = USB-C connector
- 6 = Android smartphone with installed R&S Power Viewer Mobile

Set up as shown in [Figure 3-3](#).

1. Connect the R&S NRP-ZKC cable to the power sensor. See ["To connect a cable to the host interface of the power sensor"](#) on page 15.
2. Connect the R&S NRP-ZKC cable to the Android smartphone/tablet with USB-C connector.
3. **NOTICE!** Incorrectly connecting or disconnecting the power sensor can damage the power sensor or lead to erroneous results. Ensure that you connect or disconnect the power sensor as described in [Chapter 3.4, "Connecting to a DUT"](#), on page 13.
Connect the power sensor to the signal source.
4. On the smartphone/tablet, start a software application to view the measurement results. See [Chapter 5, "Operating concepts"](#), on page 22.

3.7.3 Base unit

As a controlling host, you can use an R&S NRX base unit.



You can also operate the power sensor using other supported Rohde & Schwarz instruments with a sensor connector. For details, see also the user manual of the instrument.

- ▶ Establish the connection with the base unit using:
Host interface, see [Chapter 5.5, "R&S NRX"](#), on page 27.

4 Power sensor tour

This chapter provides an overview of the available connectors and LEDs of the power sensor.

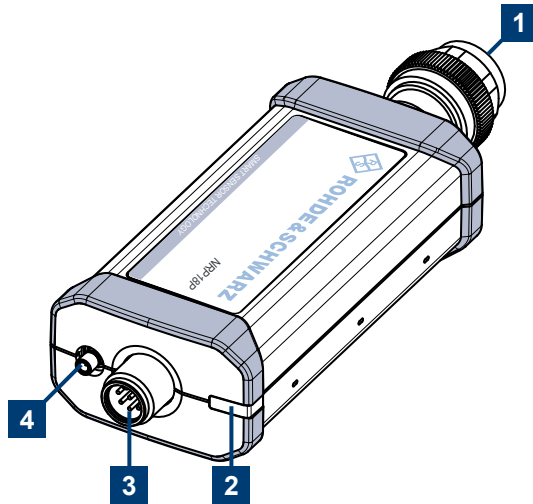


Figure 4-1: R&S NRPxxP power sensors (example)

- 1 = RF connector, see [Chapter 4.1, "RF connector"](#), on page 20
- 2 = Status LED, see [Chapter 4.2, "Status information"](#), on page 21
- 3 = Host interface connector, see [Chapter 4.3, "Host interface"](#), on page 21
- 4 = Trigger I/O connector, see [Chapter 4.4, "Trigger I/O connector"](#), on page 21

The RF relevant information is printed on the casing. The RF connector information is color-coded to distinguish the different pulse power sensor models quickly. See [Table 4-1](#).

4.1 RF connector

The RF connector is used for connecting the power sensor to a device under test (DUT) or a signal generator. See [Chapter 3.4, "Connecting to a DUT"](#), on page 13.

For maximum measurement accuracy, tighten the RF connector using a torque wrench with a nominal torque as specified in the following table.

Table 4-1: R&S NRPxxP RF connector characteristics

Power sensor	Male connector	Matching female connector	Tightening torque	Color code
R&S NRP18P	N	N	1.36 Nm (12" lbs)	Red
R&S NRP40P	2.92 mm	3.50 mm/ 2.92 mm/ SMA	0.90 Nm (8" lbs)	Yellow
R&S NRP50P	2.4 mm	2.4 mm/ 1.85 mm		Green

4.2 Status information

The status LED gives information about the state of the power sensor. The following states are defined:

Indication		State
○	White	Idle state. The power sensor performs no measurement and is ready for use.
○	Flashing white	Firmware update is in progress
○	Slow flashing white	Sanitizing in progress
●	Yellow	Wait for trigger state
●	Green	Measuring state
●	Turquoise blue	Zeroing is in progress
●	Slow flashing red	Static error You can query the error type with <code>SYSTem:SERRor?</code> .
●	Fast flashing red	Critical static error You can query the error type with <code>SYSTem:SERRor?</code> . Note: If this state occurs after a firmware update, the update was not successful. Perform the firmware update again. See also Chapter 11.3, "Problems during a firmware update" , on page 140.

4.3 Host interface

The host interface is used for establishing a connection between the power sensor and a USB host. For this purpose, an external cable is needed. See [Chapter 3.6, "Connecting a cable to the host interface"](#), on page 14.

4.4 Trigger I/O connector

The trigger I/O is a connector of SMB type.

It is used as an input for signals if the trigger source parameter is set to `EXTeRnal2`. It is used as an output for trigger signals if the power sensor is operated in the trigger sender mode.

Further information:

- [Chapter 8.5.2, "Triggering"](#), on page 43

5 Operating concepts

For operating the power sensor, you can choose from various possibilities:

- [Chapter 5.2, "Remote control"](#), on page 24
- [Chapter 5.3, "R&S NRPV"](#), on page 25
- [Chapter 5.4, "R&S Power Viewer"](#), on page 26
- [Chapter 5.5, "R&S NRX"](#), on page 27

5.1 R&S NRP Toolkit



Before you start using the power sensor, we recommend to install the R&S NRP Toolkit.

The R&S NRP Toolkit is the basic software package that supplies low-level drivers and tools for all power sensors. The components of the R&S NRP Toolkit depend on the operating system.

5.1.1 Versions and downloads

The R&S NRP Toolkit is available for:

- Microsoft Windows operating systems, as listed in [Chapter 5.1.2, "System requirements"](#), on page 22
- Linux distributions
- macOS

Several R&S NRP Toolkit versions are available on your documentation CD-ROM. The latest version for Windows is available at www.rohde-schwarz.com/software/nrp-toolkit.

To obtain an R&S NRP Toolkit for an operating system other than Microsoft Windows, contact the Rohde & Schwarz customer support: customersupport@rohde-schwarz.com

5.1.2 System requirements

Hardware requirements:

- Desktop computer or laptop, or an Intel-based Apple Mac

Supported Microsoft Windows versions:

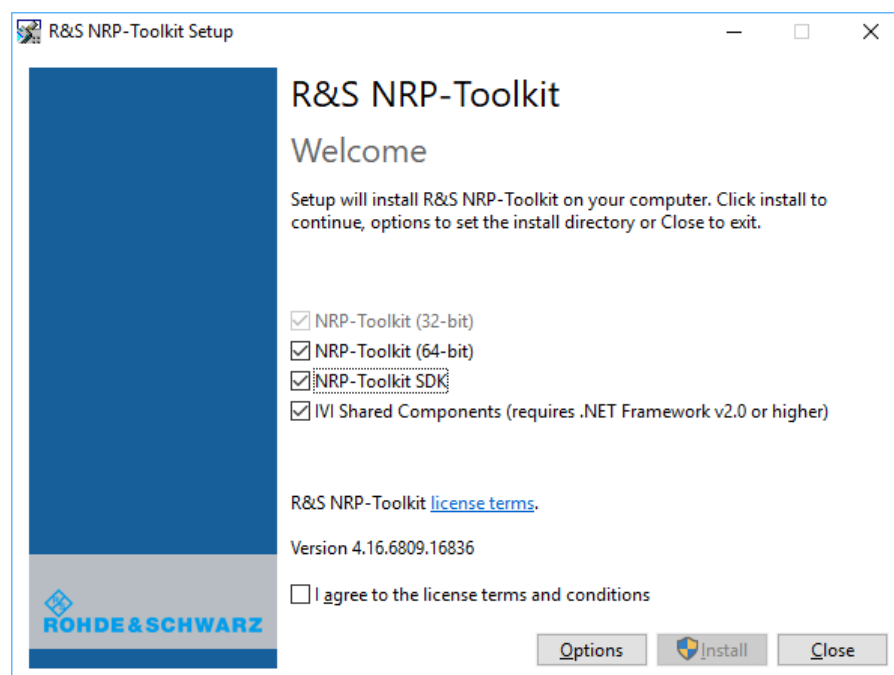
- Microsoft Windows 10 32/64-bit

5.1.3 R&S NRP Toolkit for Windows

The R&S NRP Toolkit installer for Windows-based systems contains the components described in the release notes available at www.rohde-schwarz.com/software/nrp-toolkit.

To install the R&S NRP Toolkit

1. Start the R&S NRP Toolkit installer on the Windows-based computer.
In the "NRP-Toolkit Setup" dialog, the correct R&S NRP Toolkit version for your operating system, 32-bit or 64-bit, is already selected.
2. Enable the components you want to install.
 - "NRP-Toolkit (SDK)"
The software development kit (SDK) provides programming examples for the R&S power sensors.
See [Chapter 9, "Performing measurement tasks - programming examples"](#), on page 121.
 - "IVI Shared Components"
Installs the USBTMC driver. Enabled by default because the installation is recommended.
See also [Table 10-1](#).



3. Accept the license terms to continue with the installation.
4. Click "Next" and complete the installation process.

5.1.3.1 Components of the R&S NRP Toolkit

Access: "Start" > "NRP-Toolkit"

The following tools are part of the R&S NRP Toolkit for Windows.

Configure Network Sensor

Useful if you have troubles establishing a LAN connection with an R&S NRP LAN power sensor. The tool provides the following functions:

- Configuring the network settings by (temporary) connecting the selected power sensor to the computer using USB.
- Discovering the power sensors that have been configured via the Zeroconf (APIA) protocol.

The tool comes with a guide (PDF) that is also available in the "Start" menu. The guide explains the network setup.

Firmware Update

You can use the Firmware Update for NRP Family program to load new firmware for the power sensors.

See [Chapter 6, "Firmware update"](#), on page 29.

NRP Version Display

Displays version information of all installed, power measurement-relevant software packages.

R&S NRP-Z Uncertainty Calculator

Determines the expanded measurement uncertainty. The tool comes with a manual (PDF) that is also available in the "Start" menu.

S-Parameter Update Multi

Helps loading an S-parameter table into the power sensor.

See [Chapter 8.7.4.5, "Using the S-Parameters program"](#), on page 84.

Terminal

Low-level communication program for sending commands to the power sensor.

5.2 Remote control

You can remote control the R&S NRPxxP easily. The change to remote control occurs "on the fly" and has no influence on the manual operation.

Further information:

- [Chapter 8, "Remote control commands"](#), on page 34

- [Chapter 10, "Remote control basics"](#), on page 126
- [Chapter 10.1, "Remote control interfaces and protocols"](#), on page 126
- [Chapter 3.7.1, "Computer"](#), on page 15

5.3 R&S NRPV

The R&S NRPV enables you to measure power in all available measurement modes. Also, you can use up to four power sensors simultaneously.

The R&S NRPV is provided on your documentation CD-ROM and on the Rohde & Schwarz website as a separate standalone installation package.

Required equipment

- R&S NRPxxP power sensor
- R&S NRP-ZKU cable or an R&S NRP-Z5 sensor hub and an R&S NRP-ZK6 cable to connect the power sensor to the computer
- Windows computer with installed:
 - R&S NRP Toolkit version 4.23 or higher
 - R&S NRPV version 3.2 or higher (refer to the operating manual of the R&S NRPV for a description of the installation process)

Setup

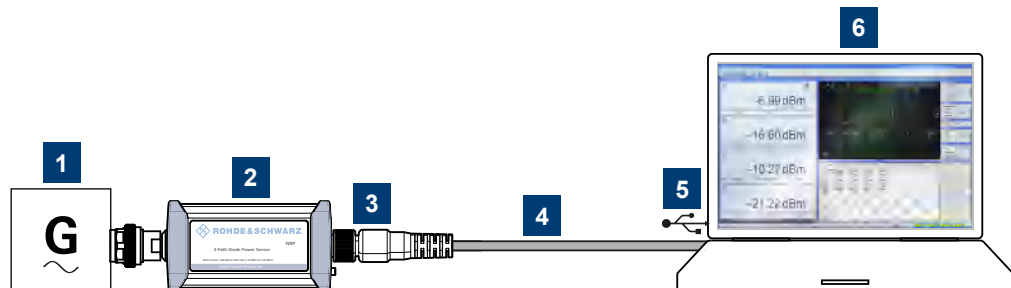


Figure 5-1: Setup with an R&S NRPV

- 1 = Signal source
- 2 = R&S NRPxxP power sensor
- 3 = Host interface connector
- 4 = R&S NRP-ZKU cable
- 5 = USB connector
- 6 = Computer with installed R&S NRPV

1. **NOTICE!** Incorrectly connecting or disconnecting the power sensor can damage the power sensor or lead to erroneous results. Ensure that you connect or disconnect the power sensor as described in [Chapter 3.4, "Connecting to a DUT"](#), on page 13.

Connect the power sensor to the signal source.

2. Connect the power sensor to the computer as shown in [Figure 5-1](#).
For a detailed description, refer to [Chapter 3.7.1.1, "USB connection"](#), on page 15.

Starting a measurement

For a detailed description of how to measure in this setup, refer to the operating manual of the R&S NRPV.

1. Start the R&S NRPV.
2. Execute zeroing.
Note: Turn off all measurement signals before zeroing. An active measurement signal during zeroing causes an error.
3. Switch on the test signal of the signal source.
4. Start a measurement.

5.4 R&S Power Viewer

The R&S Power Viewer is software that simplifies many measurement tasks. It is provided on your documentation CD-ROM and on the Rohde & Schwarz website as a separate standalone installation package.

Required equipment

- R&S NRPxxP power sensor
- R&S NRP-ZKU cable or an R&S NRP-Z5 sensor hub and an R&S NRP-ZK6 cable to connect the power sensor to the computer
- Computer with installed:
 - R&S NRP Toolkit version 4.23 or higher
 - R&S Power Viewer version 9.2 or higher (refer to the operating manual of the R&S Power Viewer for a description of the installation process)

Setup

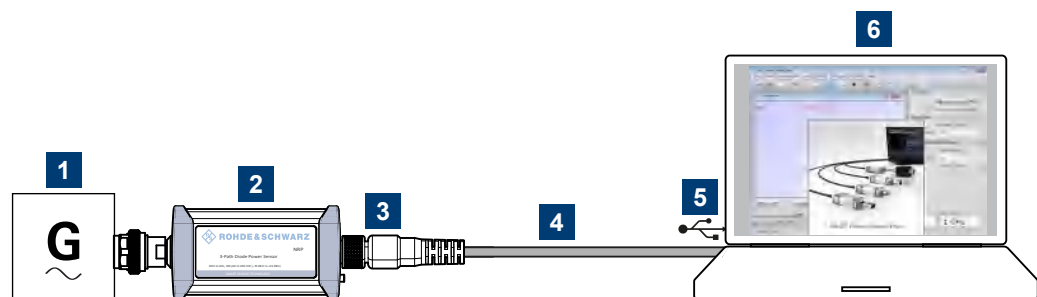


Figure 5-2: Setup with the R&S Power Viewer

- 1 = Signal source
- 2 = R&S NRPxxP power sensor
- 3 = Host interface connector
- 4 = R&S NRP-ZKU cable
- 5 = USB connector
- 6 = Computer with installed R&S Power Viewer

1. **NOTICE!** Incorrectly connecting or disconnecting the power sensor can damage the power sensor or lead to erroneous results. Ensure that you connect or disconnect the power sensor as described in [Chapter 3.4, "Connecting to a DUT"](#), on page 13.
Connect the power sensor to the signal source.
2. Connect the cables as shown in [Figure 5-2](#).
For a detailed description, refer to [Chapter 3.7.1.1, "USB connection"](#), on page 15.

Starting a measurement

For a detailed description, refer to the operating manual of the R&S Power Viewer. The manual is installed automatically during the installation of the R&S Power Viewer.

1. Start the R&S Power Viewer.
2. Execute zeroing.
Note: Turn off all measurement power signals before zeroing. An active measurement signal during zeroing causes an error.
3. Switch on the test signal of the signal source.
4. Select a measurement.
5. Start the measurement.

5.5 R&S NRX

In a measurement, the R&S NRX uses all power sensor-dependent measurement functions and displays the results. Thus, you can configure both the measurement and the power sensor.

Required equipment

- R&S NRPxxP power sensor
- R&S NRP-ZK8 cable to connect the power sensor to the R&S NRX
- R&S NRX

Setup

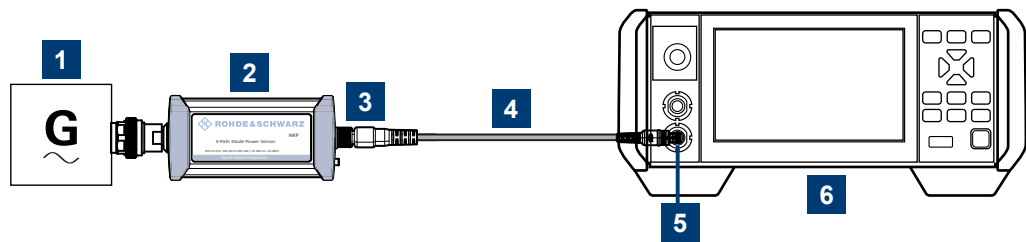


Figure 5-3: Setup with an R&S NRX base unit

- 1 = Signal source
- 2 = R&S NRPxxP power sensor
- 3 = Host interface connector
- 4 = R&S NRP-ZK8 cable
- 5 = Sensor input connector of the R&S NRX
- 6 = R&S NRX base unit

1. **NOTICE!** Incorrectly connecting or disconnecting the power sensor can damage the power sensor or lead to erroneous results. Ensure that you connect or disconnect the power sensor as described in [Chapter 3.4, "Connecting to a DUT"](#), on page 13.

Connect the power sensor to the signal source.

2. Connect the cables as shown in [Figure 5-3](#).

Starting a measurement

For a detailed description of how to measure in this setup, refer to the user manual of the R&S NRX.

1. Preset the R&S NRX and the connected R&S power sensors.
 - a) Press the [Preset] key.
 - b) Tap "Preset".
All parameters are set to their defaults.
2. **Note:** Turn off all measurement signals before zeroing. An active measurement signal during zeroing causes an error.
 - a) Switch off the power of the signal source.
 - b) Press the [Zero] key of the R&S NRX.
 - c) Tap "Zero All Sensors".
3. Configure the measurement.
 - a) In the "Measurement Settings" dialog, select the "Measurement Type", for example "Continuous Average".
 - b) Tap "Quick Setup" > "Auto Set".
4. Switch on the signal source.
The measurement starts, and the result is displayed in dBm.
5. If necessary, perform further settings.

6 Firmware update

- [Preparing the update](#).....29
- [Updating the firmware](#).....29

6.1 Preparing the update

Firmware of R&S power sensors generally has an *.rsu extension, RSU meaning Rohde & Schwarz update.

To download the update file

1. Download the most recent firmware version from the Rohde & Schwarz homepage on the Internet, since the CD-ROM accompanying the power sensor contains the firmware dating from the time of delivery. The latest firmware update files are available at:
www.rohde-schwarz.com/firmware/nrp-p
2. If the *.rsu file is packed in a *.zip archive, extract it.
3. Save the *.rsu file on the computer.

6.2 Updating the firmware

Do not interrupt the firmware update because an interruption can lead to missing or faulty firmware. Take special care not to disconnect the power supply while the update is in progress. Interrupting the power supply during the firmware update most likely leads to an unusable power sensor that needs to be sent in for maintenance.

You can choose from several methods to update the firmware installed on the power sensor.

6.2.1 Using the Firmware Update for NRP Family program

Firmware Update for NRP Family is part of the R&S NRP Toolkit. See also [Chapter 5.1, "R&S NRP Toolkit"](#), on page 22.



You can perform a firmware update with Firmware Update for NRP Family only if the power sensor is recognized as a VISA device.

To check the prerequisites

1. Ensure that a recent VISA software is installed on the computer.

2. Ensure that the R&S NRP Toolkit for Windows is installed on the computer. See [Chapter 5.1, "R&S NRP Toolkit"](#), on page 22.

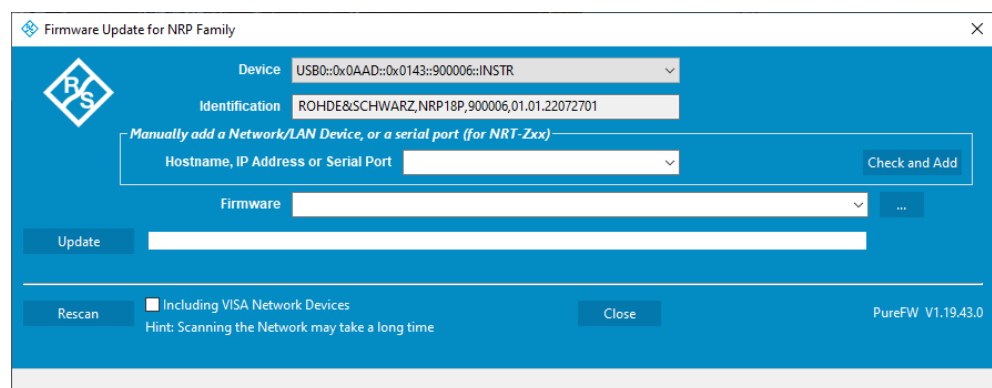
To update the firmware

A firmware update can take up to 5 minutes. Ensure that the update is not interrupted.

1. Ensure that the prerequisites are fulfilled. See ["To check the prerequisites"](#) on page 29.
2. Connect the power sensor to the computer as described in [Chapter 3.7.1, "Computer"](#), on page 15.
3. Start the Firmware Update for NRP Family program:
"Start" menu > "NRP-Toolkit" > "Firmware Update".

The program automatically starts scanning for R&S power sensors connected via USB.

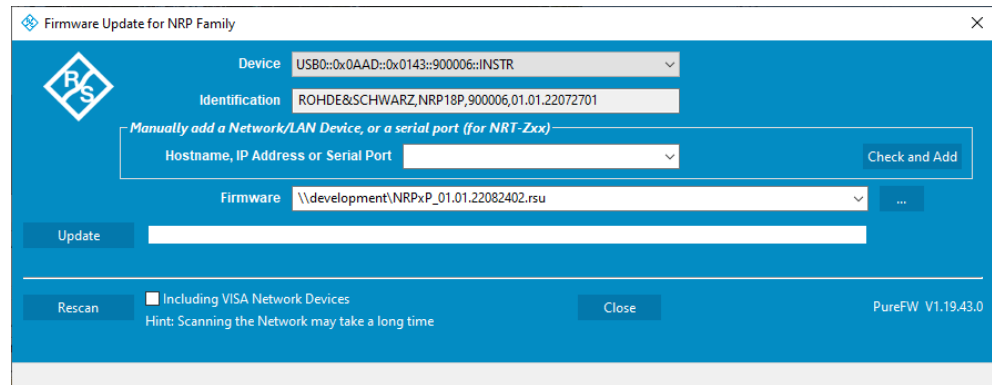
When the scan is completed, all recognized power sensors are listed under "Device".



4. If the power sensor you want to update is not listed, perform one of the following actions:
 - a) Click "Rescan" to search for attached power sensors.
 - b) Check whether all necessary drivers are installed on the computer.
For example, if the VISA library is not installed on the computer, no VISA power sensor is accessible.
See also ["Troubleshooting"](#) on page 31.
5. Under "Device", select the power sensor you want to update.
Note: The "Hostname, IP Address or Serial Port" field is not used during this procedure. Therefore, leave it empty.
6. Under "Firmware", enter the full path and filename of the update file, or press the browse button next to the field.
7. Click "Update".

During the update process, a progress bar is displayed. The update sequence can take a couple of minutes, depending on the power sensor model and the size of the selected file.

- Check if the update was successful. The firmware version in the "Identification" field must match the version you selected in the "Firmware" field.



Troubleshooting

You do not find the power sensor in the list of power sensors provided by Firmware Update for NRP Family.

The driver assigned to the power sensor is the legacy driver.

- ▶ Install a recent VISA software.

The power sensor is highlighted by a yellow exclamation mark in the Windows device manager.

Windows tries in vain to find a USB driver for the power sensor.

- ▶ Install a recent VISA software.

Further information:

- [Chapter 11.3, "Problems during a firmware update"](#), on page 140

6.2.2 Using remote control

If you want to integrate a firmware update function in an application, use `SYSTEM:FWUpdate` on page 113.

Example:

You want to update your R&S NRP40P with the `NRPxxP_02.30.21062301.rsu` file. This file has a size of 10242884 bytes.

To send the file to the power sensor for updating the firmware, your application has to assemble a memory block containing:

```
SYST:FWUP <block_data>
```

The `<block_data>` is definite length-arbitrary block data as described in [SYSTem: FWUPdate](#) on page 113.

The size of the file is 10242884. This number has 8 digits. Thus, the `<block_data>` consist of the following:

- #
- 8
How many digits follow to specify the file size.
- 10242884
Number that specifies the file size.
- <file_contents>
Contents of the *.rsu file, byte-by-byte
- 0x0a
Delimiter

In this example, you write exactly 10242905 bytes to the power sensor, for example by using a 'viWrite()' function.

The 10242905 bytes result from the values of the list above:

$$9 + 1 + 1 + 1 + 1 + 8 + 10242884 + 1$$

In a (pseudo) string notation, the memory block looks as follows:

```
SYST:FWUP #810242884<file_contents>0x0a,
```


7 Replacing an R&S NRP-Zxx with an R&S NRPxxP

The R&S NRPxxP power sensors are compatible with the R&S NRP-Zxx series of power sensors.

R&S NRPxx power sensor	Replaces R&S NRP-Zxx power sensor
R&S NRP18P	R&S NRP-Z81
R&S NRP40P	R&S NRP-Z85
R&S NRP50P	R&S NRP-Z86



To use the new power sensors, it can be required to update the drivers. For computer-based software applications (R&S NRPV and R&S Power Viewer), install the latest R&S NRP Toolkit (version 4.23 or higher).

For using the power sensors with base units, signal generators, spectrum analyzers or other Rohde & Schwarz instruments, install the latest firmware version.

7.1 Important difference

After powering the R&S NRPxxP, the firmware is loaded. Because the USB power sensors are plug-and-play devices, the drivers are loaded after enumeration in the host system. For the R&S NRP-Zxx power sensors, the whole process takes 4 s to 5 s. For USB power sensors, the process can take up to 8 s.

Otherwise, the R&S NRP-Zxx power sensors and the USB power sensors are compatible as far as possible.

7.2 Prerequisites

Install the R&S NRP Toolkit version 4.23 or higher, see [Chapter 5.1, "R&S NRP Toolkit"](#), on page 22.

The new version of the R&S NRP Toolkit is compatible with both the R&S NRP-Zxx and the R&S NRPxxP so that its installation does not affect the usage of the R&S NRP-Zxx.

After the new version of the R&S NRP Toolkit is installed, you can connect the R&S NRPxxP to the computer and use it with Rohde & Schwarz software applications or your own programs. For information on Rohde & Schwarz software applications, see the release notes and the manual of the software application.

8 Remote control commands

In the following, all implemented commands are listed according to the command system and then described in detail. Mostly, the notation used complies with SCPI specifications.

8.1 Conventions used in SCPI command descriptions

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 NRPxxP 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.

Units

For physical quantities, you can enter the unit. Only basic units are allowed and recognized.

Table 8-1: Units

Basic unit	Also noted as
Hz	Frequency or HZ
s	Seconds
W	Watts

Basic unit	Also noted as
degrees	Angle
PCT	Percent
dB	DB
dBm	DBM
dBuV	DBUV

8.2 Notations

For a detailed description of SCPI notations, see [Chapter 10, "Remote control basics"](#), on page 126.

Numeric suffixes <n>

If a command can be applied to multiple instances of an object, e.g. specific power sensors, the required instances can be specified by a suffix added to the command. Numeric suffixes are indicated by angular brackets (<1...4>, <n>, <I>) and are replaced by a single value in the command. Entries without a suffix are interpreted as having the suffix 1.

Optional keywords []

Some command systems permit certain keywords to be inserted into the header or omitted. These keywords are marked by square brackets in the description. The power sensor must recognize the long command to comply with the SCPI standard. Some commands are considerably shortened without these optional mnemonics.

So you can use the short or long form for the commands, distinguished here by uppercase and lowercase letters. Also, you can shorten by omitting optional keywords.

Example:

Command [SENSe<Sensor>:] [POWer:] [AVG:] SMOothing:STATe 1 can be written as:

```
SENSe1:POWer:AVG:SMOothing:STATe 1
SENS:POW:AVG:SMO:STAT 1
SENSe:POWer:SMOothing:STATe 1
SENSe:SMOothing:STATe 1
SMOothing:STATe 1
SMO:STAT 1
```

Parameters

Parameters must be separated from the header by a "white space". If several parameters are specified in a command, they are separated by a comma (,).

Example:

Definition: [SENSE<Sensor>:]AVERage:COUNT:AUtO:NSRatio <nsr>

Command: AVER:COUN:AUtO:NSR 0.01

Special characters | and { }

	<p>A vertical bar in parameter definitions indicates alternative possibilities in the sense of "or". The effect of the command differs, depending on which parameter is used.</p> <p>Example:</p> <p>Definition: INITiate:CONTinuous ON OFF</p> <p>Command INITiate:CONTinuous ON starts the measurements</p> <p>Command INITiate:CONTinuous OFF stops the measurements</p>
{ }	Parameters in braces can be included in the command once, several times or not at all.

8.3 Common commands

The common commands are taken from the IEEE 488.2 (IEC 625–2) standard. The headers of these commands consist of an asterisk * followed by three letters.

*CLS.....	36
*ESE.....	37
*ESR?.....	37
*IDN?.....	37
*IST?.....	37
*OPC.....	37
*OPT?.....	38
*PRE.....	38
*RCL.....	38
*RST.....	38
*SAV.....	38
*SRE.....	39
*STB?.....	39
*TRG.....	39
*TST?.....	39
*WAI.....	39

***CLS**

Clear status

Resets the following:

- Status byte (STB)
- Standard event register (ESR)
- EVENT part of the QUESTionable and the OPERation register
- Error/event queue

The command does not alter the `ENABLE` and `TRANSITION` parts of the registers.

Usage: Event

***ESE <register>**

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:

<register>	Range:	0 to 255
	*RST:	0

***ESR?**

Event status read

Returns the contents of the event status register in decimal form (0 to 255) and then sets the register to zero.

Usage: Query only

***IDN?**

Identification

Returns a string containing information on the identity of the power sensor (device identification code). In addition, the version number of the installed firmware is indicated.

Usage: Query only

***IST?**

Individual status

Returns the current value of the IST flag in decimal form. The IST flag is the status bit which is sent during a parallel poll.

Usage: Query only

***OPC**

Operation complete

Sets bit 0 in the event status register when all preceding commands have been executed. Send this command at the end of a program message. It is important that the read timeout is set sufficiently long.

The query always returns 1 because the query waits until all previous commands are executed.

*OPC? basically functions like *WAI, but also returns a response. The response is an advantage, because you can query the execution of commands from a controller program before sending new commands. Thus preventing overflow of the input queue when too many commands are sent that cannot be executed.

*OPT?

Option identification

Returns a comma-separated list of installed options.

Usage: Query only

*PRE <register>

Parallel poll register enable

Sets the parallel poll enable register to the specified value or queries the current value.

Parameters:

<register>	Range:	0 to 255
	*RST:	0

*RCL <number>

Recall

Calls the device state which has been stored with the *SAV command under the specified number.

Setting parameters:

<number>	Range:	0 to 9
	*RST:	0

Usage: Setting only

*RST

Reset

Sets the instrument to a defined default status. The default settings are indicated in the description of commands.

The command corresponds to the `SYSTem:PRESet` command.

Usage: Event

*SAV <number>

Save

Stores the current device state under the specified number.

Setting parameters:

<number> Range: 0 to 9
 *RST: 0

Usage: Setting only

***SRE <register>**

Service request enable

Sets the service request enable register to the specified value. This command determines under which conditions a service request is triggered.

Parameters:

<register> Range: 0 to 255
 *RST: 0

***STB?**

Status byte

Returns the contents of the status byte in decimal form.

Usage: Query only

***TRG**

Trigger

Triggers a measurement if the following conditions are met:

- Power sensor is in the waiting for trigger state.
- Trigger source is set to BUS.

See [TRIGger:SOURce](#) BUS.

Usage: Event

***TST?**

Selftest

Triggers a selftest of the R&S NRPxxP and outputs the result. 0 indicates that no errors have occurred.

Usage: Query only

***WAI**

Wait to continue

Prevents the execution of the subsequent commands until all preceding commands have been executed and all signals have settled.

Usage: Event

8.4 Preparing for the measurement

Before starting a measurement, you need to do the following:

- [Selecting a measurement mode](#)..... 40
- [Configuring the measured values](#)..... 41

8.4.1 Selecting a measurement mode

- ▶ Before starting a measurement, select the measurement mode using:

`SENSe<Sensor>:FUNCTION`

The available measurement modes and how to configure them are described in [Chapter 8.6, "Configuring the measurement modes"](#), on page 56.

SENSe<Sensor>:FUNCTION <function>

Sets the measurement mode.

Parameters:

<function>

"POWER:AVG"

Continuous average mode

See [Chapter 8.6.1, "Continuous average measurement"](#), on page 56.

"POWER:BURSt:AVG"

Burst average mode

See [Chapter 8.6.2, "Burst average measurement"](#), on page 59.

"POWER:TSLot:AVG"

Timeslot mode

See [Chapter 8.6.3, "Timeslot measurement"](#), on page 60.

"XTIME:POWER"

Trace mode

See [Chapter 8.6.4, "Trace measurement"](#), on page 62.

Pulse analysis

See [Chapter 8.6.5, "Pulse analysis measurement"](#), on page 67.

"XPOWER:CCDFunction"

Statistics (CCDF) mode

See [Chapter 8.6.6, "Statistics measurement"](#), on page 70.

"XPOWER:PDFunction"

Statistics (PDF) mode

See [Chapter 8.6.6, "Statistics measurement"](#), on page 70.

*RST: "POWER:AVG"

8.4.2 Configuring the measured values

Before starting a measurement, you can configure the measurand or enable the measurement of additional-measured values.

CALCulate:FEED.....	41
SENSe<Sensor>:AUXiliary.....	42

CALCulate:FEED <mode>

If you query measurement data using `FETCh<Sensor>[:SCALar][:POWer][:AVG]?`, the power sensor returns data of the measurand that was configured before. Generally, this measurand is the average power. However, the power sensor can also output data of other measurands.

To configure which measurand the `FETCh<Sensor>[:SCALar][:POWer][:AVG]?` command reads, use the `CALCulate:FEED` command before the measurement is initiated. Depending on the measurement mode, the following settings are possible:

SENS:FUNC	Possible CALC:FEED	Meaning
"POWer:AVG"	"POWer:AVERAge"	Average value
	"POWer:PEAK"	Peak value
	"POWer:RANDom"	Randomly selected value from the measurement interval
"POWer:BURSt:AVG"	"POWer:AVERAge"	Average value
	"POWer:PEAK"	Peak value
	"POWer:RANDom"	Randomly selected value from the measurement interval
"POWer:TSLot:AVG"	"POWer:AVERAge"	Average value
	"POWer:PEAK"	Peak value
	"POWer:RANDom"	Randomly selected value from the measurement interval
"XTIME:POWer"	"POWer:TRACe"	Measurement sequence
	"POWer:PEAK:TRACe"	Peak value of the samples per trace point
	"POWer:RANDom:TRACe"	Randomly selected value of the samples per trace point
"XPoWer:CCDFunction"	CCDF:TRACe	Complementary cumulative distribution function
"XPoWer:PDFunction"	PDF:TRACe	Probability density function

Parameters:

<mode> *RST: "POWer:AVERAge"

SENSe<Sensor>:AUXiliary <mode>

Enables the measurement of additional-measured values that are determined together with the main-measured value.

Parameters:

<mode> NONE | MINMax | RNDMax

NONE

No additional values are measured.

MINMax

Minima and maxima of the trace are transmitted together with the measured value.

Usually, extreme values are lost due to averaging the measured values.

RNDMax

Not available in timeslot mode.

Randomly selected samples are transmitted. All evaluations use these values instead of the average values.

*RST: NONE

8.5 Controlling the measurement

The power sensor offers a bunch of possibilities to control the measurement:

- Do you want to start the measurement immediately after the initiate command or do you want to wait for a trigger event?
- Do you want to start a single measurement cycle or a sequence of measurement cycles?
- Do you want to output each new average value as a measurement result or do you want to bundle more measured values into one result?

8.5.1 Starting and ending a measurement

ABORt.....	42
INITiate:ALL.....	43
INITiate[:IMMEDIATE].....	43
INITiate:CONTinuous.....	43

ABORt

Immediately interrupts the current measurement. If the measurement has been started as a single measurement (`INITiate[:IMMEDIATE]`), the power sensor goes into the idle state. However, if a continuous measurement is in progress (`INITiate:CONTinuous ON`), the trigger system of the power sensor enters the waiting for trigger state. When the trigger condition is met, a new measurement is immediately started.

See also [Chapter 8.5, "Controlling the measurement"](#), on page 42.

Usage: Event

**INITiate:ALL
INITiate[:IMMediate]**

Starts a single measurement cycle. The power sensor changes from the idle state to the waiting for trigger state. When the trigger condition is fulfilled, the power sensor begins the measurement. Depending on the number of trigger events that are required, e.g. for averaging, the power sensor enters the waiting for trigger state several times. Once the entire measurement is completed, a measurement result is available, and the power sensor enters the idle state again.

Use this command only after the continuous measurement mode has been disabled using `INITiate:CONTInuous OFF`.

See also [Chapter 8.5, "Controlling the measurement"](#), on page 42.

Example: See [Chapter 9.2, "Performing a buffered continuous average measurement"](#), on page 121.

Usage: Event

INITiate:CONTInuous <state>

Enables or disables the continuous measurement mode. In continuous measurement mode, the power sensor does not go into the idle state after a measurement has been completed, but immediately executes another measurement cycle.

See also [Chapter 8.5.2, "Triggering"](#), on page 43.

Parameters:

<state>

ON

Measurements are performed continuously. If a measurement is completed, the power sensor does not return to the idle state but enters the waiting for trigger state again.

OFF

Ends the continuous measurement mode, and sets the power sensor to the idle state.

*RST: OFF

Example: See [Chapter 9.2, "Performing a buffered continuous average measurement"](#), on page 121.

8.5.2 Triggering

In a basic continuous measurement, the measurement is started immediately after the initiate command, see also [Chapter 8.5.2.2, "Waiting for a trigger event"](#), on page 44. However, sometimes you want that the measurement starts only if a specific condition is fulfilled. For example, if a signal level is exceeded, or in certain time intervals. For these cases, you can define a trigger for the measurement.

Further information:

- [Chapter 8.5.5, "Configuring the trigger"](#), on page 51

8.5.2.1 Trigger states

The power sensor has trigger states to define the exact start and stop time of a measurement and the sequence of a measurement cycle. The following states are defined:

- **Idle**
The power sensor performs no measurement. After powered on, the power sensor is in the idle state.
- **Waiting for trigger**
The power sensor waits for a trigger event that is defined by the trigger source. When the trigger event occurs, the power sensor enters the measuring state.
- **Measuring**
The power sensor is measuring data. It remains in this state during the measurement. When the measurement is completed, it exits this state immediately.

8.5.2.2 Waiting for a trigger event

Before a trigger can be executed, the power sensor has to be set to the waiting for trigger state. Depending on the required number of measurement cycles, you use one of the following commands:

- `INITiate:CONTinuous`
A new measurement cycle is started automatically after the previous one has been terminated.
- `INITiate[:IMMediate]`
The number of measurement cycles is restricted.
If `TRIGger:COUNT 1` is set, the command starts a single measurement cycle that renders one result. Every time you send this command, a new measurement cycle is started.
Otherwise, as many measurement cycles are performed as determined by the trigger count.

8.5.2.3 Trigger sources

The possible trigger conditions and the execution of a trigger depend on the selected trigger mode and trigger source.

If the signal power exceeds or falls below a reference level set by the trigger level, the measurement is started after the defined delay time. Waiting for a trigger event can be skipped.

Trigger source	Description	Remote commands to initiate the measurement
"Hold"	Triggered by the remote command.	<code>TRIGger:IMMediate</code>
"Immediate"	Measures immediately, does not wait for trigger condition.	-

Trigger source	Description	Remote commands to initiate the measurement
"Internal"	Uses the input signal as trigger signal.	TRIGger:IMMediate
"External 1"	Uses the digital input signal supplied using a differential pair in the 8-pin sensor cable.	TRIGger:IMMediate
"External 2"	Uses the digital input signal supplied at the SMB connector.	TRIGger:IMMediate
"Bus"	Triggered by the remote command.	*TRG TRIGger:IMMediate

8.5.2.4 Dropout time

The dropout time is useful when dealing with signals with several active slots, for example GSM signals, see [Figure 8-1](#). When measuring in sync with the signal, a trigger event is to be produced at A, but not at B or C.

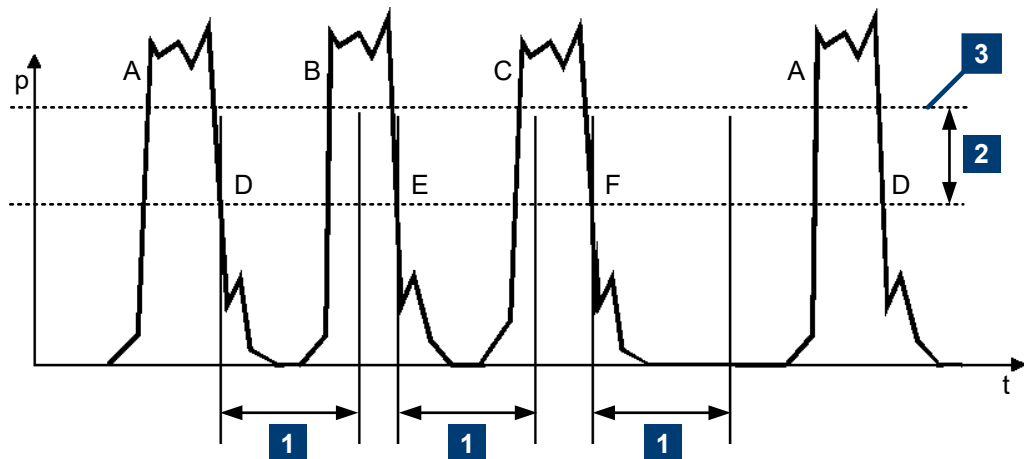


Figure 8-1: Significance of the dropout time

- 1 = Dropout time
- 2 = Trigger hysteresis
- 3 = Trigger level

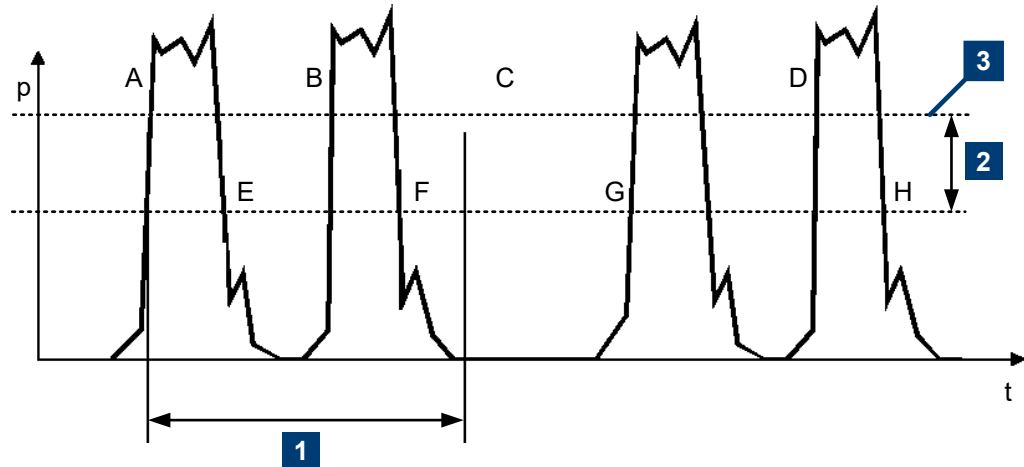
The RF power between the slots is below the threshold defined by the trigger level and the trigger hysteresis. Therefore, the trigger hysteresis alone cannot prevent triggering at B or at C. Therefore, set the dropout time greater than the time elapsed between points D and B and between E and C, but smaller than the time elapsed between F and A. Thus, you ensure that triggering takes place at A.

Because the mechanism associated with the dropout time is reactivated whenever the trigger threshold is crossed, you can obtain also unambiguous triggering for many complex signals.

If you use a hold-off time instead of a dropout time, you can obtain stable triggering conditions - regular triggering at the same point. But you cannot achieve exclusive triggering at A.

8.5.2.5 Hold-off time

During the hold-off time, a period after a trigger event, all trigger events are ignored.



- 1 = Hold-off time
- 2 = Trigger hysteresis
- 3 = Trigger level

8.5.3 Controlling the measurement results

The R&S NRPxxP can cope with the wide range of measurement scenarios with the help of the so-called "termination control". Depending on how fast your measurement results change, you can define, how the measurement results are output.

In continuous average mode, use `SENSe<Sensor>:AVERage:TCONtrol`.

Repeating termination control

Outputs a measurement result when the entire measurement has been completed. This means that the number of measurement cycle repetitions is equal to the set average count. If the average count is large, the measurement time can be very long.

Useful if you expect slow changes in the results, and you want to avoid outputting redundant data.

Moving termination control

Outputs intermediate values to facilitate early detection of changes in the measured quantity. This means that for each partial measurement, a new average value is output as a measurement result. Thus, the measurement result is a moving average of the last partial measurements. How many of the partial measurements are averaged is defined by the average count.

Useful if you want to detect trends in the result during the measurement.

8.5.4 Interplay of the controlling mechanisms

In the following examples, continuous measurement scenarios are used. But these scenarios apply also to single measurements. The only difference is that a single measurement is not repeated.

8.5.4.1 Continuous average mode

General settings for these examples:

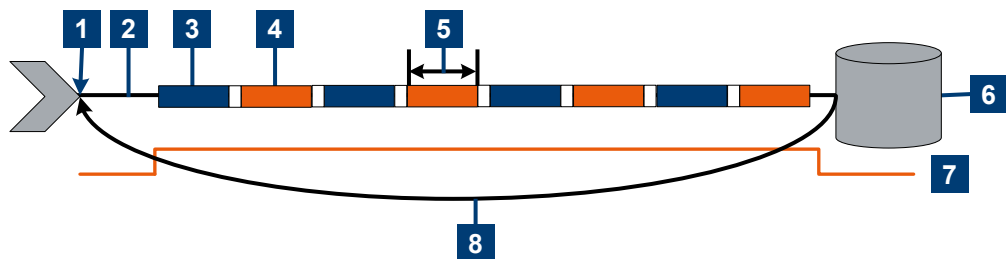
- `INITiate:CONTinuous ON`
- `SENSe<Sensor>:AVERage:COUNT 4`
- `SENSe<Sensor>:AVERage:COUNT:AUTO OFF`

Example: Repeating termination control

Further settings for this example:

- `SENSe<Sensor>:AVERage:TCONtrol REPEAT`

The measurement is started by the trigger event. Due to the chopper phases, one measurement lasts twice the defined aperture time. As defined by the average count, after 4 measurements, the result is averaged and available. During the whole measurement cycle, the trigger synchronization is high (`TRIGger:SYNC:STATE ON`).



- 1 = Start of the measurement cycle
- 2 = Trigger event
- 3 = Noninverted chopper phase
- 4 = Inverted chopper phase
- 5 = Duration of one aperture time ($1 \times t_{AP}$) \triangleq length of one chopper phase
- 6 = Measurement result
- 7 = Trigger synchronization
- 8 = Return to the start of the measurement cycle

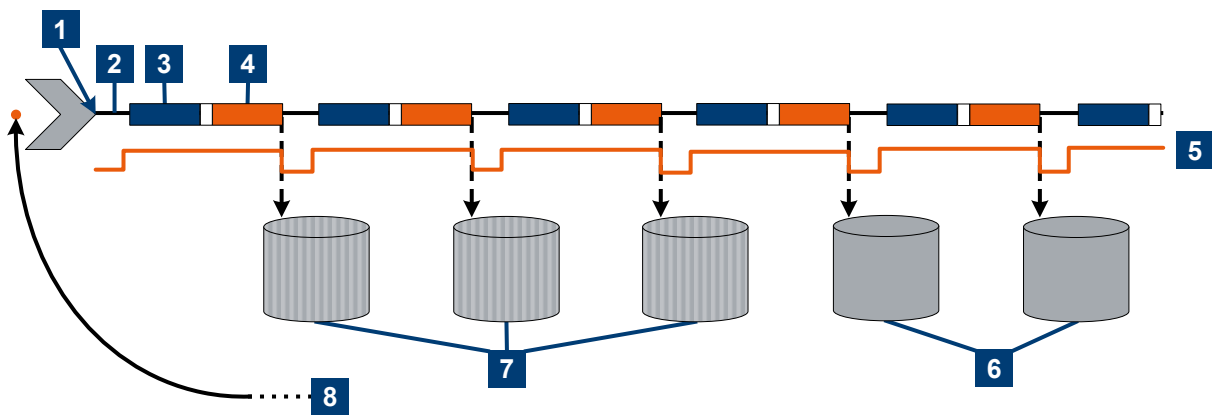
Example: Moving termination control

Further settings for this example:

- `SENSe<Sensor>:AVERAge:TCONtrol MOVing`
- `TRIGger:COUnT 16`

Every measurement is started by a trigger event. Due to the chopper phases, one measurement lasts twice the defined aperture time. During each measurement, the trigger synchronization is high (`TRIGger:SYNC:STATe ON`). Every measurement provides a result. During the settling phase, the amount of the result is already correct, but the noise is higher. After 4 measurements, when the average count is reached, settled data is available.

When the trigger count is reached (`TRIGger:COUnT` on page 52), the power sensor returns to the idle state.

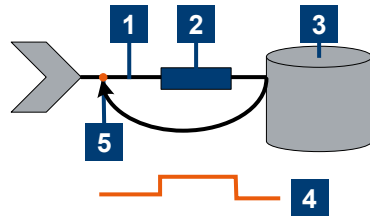


- 1 = Start of the measurement cycle
- 2 = Trigger events
- 3 = Noninverted chopper phase
- 4 = Inverted chopper phase
- 5 = Trigger synchronization
- 6 = Averaged measurement result after average count is reached
- 7 = Measurement result before average count is reached
- 8 = Return to idle state after trigger count (= 16 in this example) is reached

Example: Average count = 1

```
SENSe<Sensor>:AVERage:COUNT 1
```

For average count 1, the setting of the termination control has no impact. In both cases, the measurement runs for the duration of one aperture time. Then, settled data is available, and the power sensor returns to the idle state.



- 1 = Trigger event
- 2 = Noninverted chopper phase
- 3 = Measurement result
- 4 = Trigger synchronization
- 5 = Return to idle state

8.5.4.2 Trace mode

General settings for the first two examples:

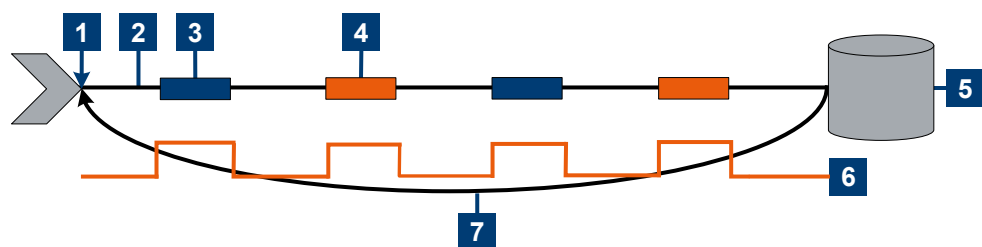
- `INITiate:CONTInuous ON`
- `SENSe<Sensor>:TRACe:AVERage:COUNT 2`
- `SENSe<Sensor>:TRACe:AVERage[:STATe] ON`

Example: Repeating termination control

Further settings for this example:

- `SENSe<Sensor>:TRACe:AVERage:TCONtrol REPEAT`

Every chopper phase is started by a trigger event and lasts the defined trace time. During a chopper phase, the trigger synchronization is high (`TRIGger:SYNC:STATe ON`). After 2 chopper phases, 1 measurement is completed. As defined by the trace average count, after 2 measurements, the trace measurement result is averaged and available.



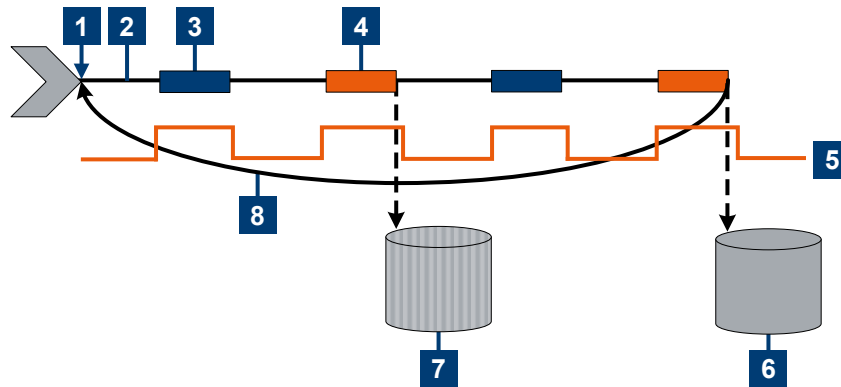
- 1 = Start of the measurement cycle
- 2 = Trigger events
- 3 = Noninverted chopper phase
- 4 = Inverted chopper phase
- 5 = Trace measurement result
- 6 = Trigger synchronization
- 7 = Return to the start of the measurement cycle

Example: Moving termination control

Further settings for this example:

- `SENSe<Sensor>:TRACe:AVERage:TCONtrol MOVing`

Every chopper phase is started by a trigger event and lasts the defined trace time. During a chopper phase, the trigger synchronization is high (`TRIGger:SYNC:STATe ON`). Every measurement provides a result. After 2 measurements, when the trace average count is reached, the settled trace data result is available.

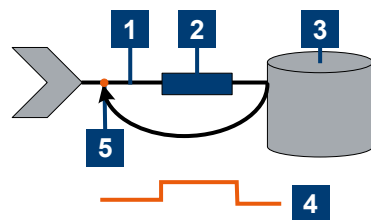


- 1 = Start of the measurement cycle
- 2 = Trigger events
- 3 = Noninverted chopper phase
- 4 = Inverted chopper phase
- 5 = Trigger synchronization
- 6 = Averaged trace data result after trace average count is reached
- 7 = Trace measurement result before average count is reached
- 8 = Return to the start of the measurement cycle

Example: Average count = 1

`SENSe<Sensor>:TRACe:AVERage:COUNT 1`

For average count 1, the setting of the termination control has no impact. In both cases, the measurement runs for the duration of one trace time. Then, settled trace data is available, and the power sensor returns to the idle state.



- 1 = Trigger event
- 2 = Noninverted chopper phase
- 3 = Trace measurement result
- 4 = Trigger synchronization
- 5 = Return to idle state

8.5.5 Configuring the trigger

Further information:

- Chapter 8.5, "Controlling the measurement", on page 42

Remote commands:

TRIGger:ATRigger:DELay.....	51
TRIGger:ATRigger:EXECuted?.....	51
TRIGger:ATRigger[:STATe].....	51
TRIGger:COUNt.....	52
TRIGger:DELay.....	52
TRIGger:DTIME.....	52
TRIGger:EXTernal<2...2>:IMPedance.....	53
TRIGger:HOLDoff.....	53
TRIGger:HYSTeresis.....	53
TRIGger:IMMediate.....	53
TRIGger:LEVel.....	54
TRIGger:LEVel:UNIT.....	54
TRIGger:SENDER:PORT.....	54
TRIGger:SENDER:STATe.....	55
TRIGger:SLOPe.....	55
TRIGger:SOURce.....	55
TRIGger:SYNC:STATe.....	56

TRIGger:ATRigger:DELay <delay>

Effective only if `TRIGger:ATRigger[:STATe]` is set to ON.

Sets the delay between the artificial trigger event and the beginning of the actual measurement

Parameters:

<delay>	Range:	0.1 to 5.0
	*RST:	0.3
	Default unit:	Seconds

TRIGger:ATRigger:EXECuted?

Queries the number of measurements that were triggered automatically since `TRIGger:ATRigger[:STATe]` was set to ON.

In normal scalar measurements, this number can only be 0 or 1. If a buffered measurement was executed, this number indicates how many results in the returned array of measurement data were executed without a real trigger event.

Usage: Query only

TRIGger:ATRigger[:STATe] <state>

Controls the automatic trigger function. If enabled, an artificial trigger is generated if the delay time has elapsed after the measurement start and no trigger event has occurred.

The delay time is set using `TRIGger:ATRigger:DElay`.

Parameters:

<state> *RST: OFF

TRIGger:COUNT <count>

Sets the number of measurement cycles to be performed when the measurement is started using `INITiate[:IMMediate]`.

This number equals the number of results that can be obtained from the power sensor after a single measurement. As long as the defined number of measurements is not executed, the power sensor automatically initiates another measurement internally when the current result is available.

This command is particularly useful in conjunction with buffered measurements. For example, to fill a buffer with a predefined size with measurements that have been triggered externally or by `*TRG` without having to start the measurement multiple times.

Parameters:

<count> Range: 1 to 2147483646
 *RST: 1

Example: See [Chapter 9.2, "Performing a buffered continuous average measurement"](#), on page 121.

TRIGger:DElay <delay>

Sets the delay between the trigger event and the beginning of the actual measurement (integration).

Parameters:

<delay> Range: -5.0 to 10.0
 *RST: 0.0
 Default unit: s

TRIGger:DTIME <dropout_time>

Sets the dropout time for the internal trigger source. During this time, the signal power must exceed (negative trigger slope) or undercut (positive trigger slope) the level defined by the trigger level and trigger hysteresis. At least, this time must elapse before triggering can occur again.

See [Chapter 8.5.2.4, "Dropout time"](#), on page 45.

Parameters:

<dropout_time> Range: 0.00 to 10.00
 *RST: 25.0e-9
 Default unit: s

TRIGger:EXternal<2...2>:IMPedance <impedance>

Effective only if **TRIGger:SOURce EXternal2** is set.

Sets termination resistance of the second external trigger input. Choose the setting that fits the impedance of the trigger source to minimize reflections on the trigger signals.

Suffix:

<2...2> 2

Parameters:

<impedance> HIGH | LOW
HIGH
~10 kΩ
LOW
50 Ω
*RST: HIGH

TRIGger:HOLDoff <holdoff>

Sets the hold-off time, see [Chapter 8.5.2.5, "Hold-off time"](#), on page 46.

Parameters:

<holdoff> Range: 0.00 to 10.00
 *RST: 0.00
 Default unit: Seconds

TRIGger:HYSTeresis <hysteresis>

Sets the hysteresis. A trigger event occurs, if the trigger level:

- Falls below the set value on a rising slope.
- Rises above the set value on a falling slope

Thus, you can use this setting to eliminate the effects of noise in the signal for the edge detector of the trigger system.

Parameters:

<hysteresis> Range: 0.00 to 10.00
 *RST: 0.00
 Default unit: DB

TRIGger:IMMediate

Causes a generic trigger event. The power sensor leaves the waiting for trigger state immediately, irrespective of the trigger source and the trigger delay, and starts the measurement.

This command is the only way to start a measurement if the trigger source is set to hold, **TRIGger:SOURce HOLD**. Only one measurement cycle is executed, irrespective of the averaging factor.

Usage: Event

TRIGger:LEVel <level>

Effective only if [TRIGger:SOURce](#) INTernal.

Sets the trigger threshold for internal triggering derived from the test signal.

If an S-parameter device and/or the offset correction are enabled, the trigger threshold is referenced to the correction data.

If the S-parameter device and/or the offset correction are disabled, the trigger threshold and its input limits are adjusted as necessary.

If you enter a value without unit, the unit is defined by [TRIGger:LEVel:UNIT](#).

Parameters:

<level>	Range:	10.0e-7 to 100.0e-3
	*RST:	1.0e-4
	Default unit:	Watts

TRIGger:LEVel:UNIT <unit>

Sets the unit of the trigger level if this value is entered without a unit. See also [TRIGger:LEVel](#) on page 54.

Parameters:

<unit>	DBM W DBUV
	*RST: W

TRIGger:SENDER:PORT <sender_port>

Selects the port where the power sensor outputs its own trigger event in case it is trigger sender. See [TRIGger:SENDER:STATE](#) for more information.

If the power sensor is the trigger sender, it can output its trigger event either on the `EXTernal<1>` or `EXTernal2`.

If the power sensor triggers itself, the trigger source of the power sensor must be assigned to the other external port, as shown in the examples.

Parameters:

<sender_port>	EXT1 EXTernal1 EXT2 EXTernal2
	*RST: EXT1

Example:

```
TRIG:SEND:PORT EXT1
TRIG:SOUR EXT2
TRIG:SEND:STAT ON
```

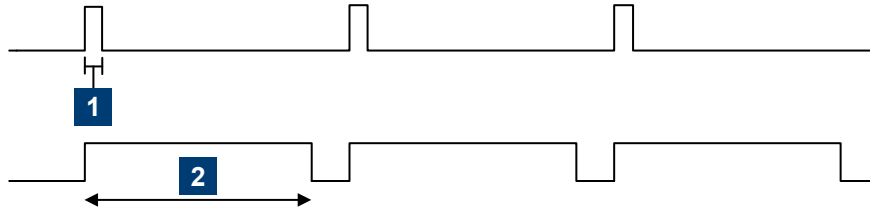
Example:

```
TRIG:SEND:PORT EXT2
TRIG:SOUR EXT1
TRIG:SEND:STAT ON
```

TRIGger:SENDER:STATe <state>

Available in trace, timeslot modes with **TRIGger:SOURce** **INTernal** or **BUS**.

Enables or disables the trigger sender mode of the power sensor. In this state, the power sensor can output a digital trigger signal in sync with its own trigger event.



1 = digital trigger signal of 1 μ s

2 = one measurement cycle

If enabled, select the output port for the trigger signal using **TRIGger:SENDER:PORT**.

Typically, the trigger sender uses its internal trigger source. But you can also trigger the trigger sender externally, because the power sensor has two external trigger connectors. If you trigger the trigger sender externally, use **EXTernal1** as external trigger input port (trigger source) and **EXTernal2** as trigger sender output port or vice versa.

Parameters:

<state> *RST: OFF

TRIGger:SLOPe <slope>

Effective only if **TRIGger:SOURce** is set to **INTernal** or **EXTernal**.

Determines which edge of the envelope power, with internal triggering, or increasing voltage, with external triggering, is used for triggering.

Parameters:

<slope> POSitive | NEGative
POSitive
 Rising edge
NEGative
 Falling edge
 *RST: POSitive

TRIGger:SOURce <source>

Selects the source for the trigger event detector.

Parameters:

<source> HOLD | IMMEDIATE | INTernal | BUS | EXTernal | EXT1 |
 EXTernal1 | EXT2 | EXTernal2
 See [Chapter 8.5.2.3, "Trigger sources"](#), on page 44.
 *RST: IMMEDIATE

TRIGger:SYNC:STATE <state>

Available in:

- Continuous average mode with `TRIGger:SOURce IMMEDIATE`
- Trace, timeslot modes with `TRIGger:SOURce INTERNAL`

Enable the synchronization state if you want to synchronize several power sensors connected to a base unit. The synchronization works as follows.

The external trigger bus is blocked as long as a power sensor is in the "measuring" state. The external trigger bus is enabled only when all power sensors are in the "waiting for trigger" or "idle" state. Thus, the power sensors start to measure simultaneously.

Make sure that the measurement time is the same for all power sensors involved in the measurement. Otherwise, the trigger bus is blocked by any power sensor that has completed its measurement before the others and has returned to the "idle" state.



1 = measurement time

During the measurement, the trigger synchronization is high. For a continuous average measurement, the measurement time is calculated as described in [Chapter 8.6.1.4, "Calculating the measurement time"](#), on page 58.

For a trace or timeslot measurement, one measurement cycle is one sweep.

For details on trigger states, see [Chapter 8.5.2.1, "Trigger states"](#), on page 44.

Parameters:

<state> *RST: OFF

8.6 Configuring the measurement modes

In the following, the settings needed for configuring a measurement mode are described.

Further information:

- [Chapter 8.7, "Configuring basic measurement parameters"](#), on page 74
- [Chapter 8.5, "Controlling the measurement"](#), on page 42
- [Chapter 8.9, "Querying measurement results"](#), on page 96

8.6.1 Continuous average measurement

The continuous average mode measures the signal average power asynchronously within definable time intervals (sampling windows). The aperture (width of the sampling windows) can be defined.



For information on querying the measurement results, see [Chapter 8.9.1, "Continuous average measurement results"](#), on page 96.

8.6.1.1 Defining the sampling window

SENSe<Sensor>:[POWER:][AVG:]APERture <integration_time>

Sets the duration of the sampling window. During this time interval, the average signal power is measured.

Parameters:

<integration_time> Range: 1.0e-6 to 1.0
 *RST: 10.0e-6
 Default unit: Seconds

8.6.1.2 Reducing noise and zero offset

The smoothing filter can reduce result fluctuations caused by modulation. But activating it increases the inherent noise of the power sensor by approx. 20 %, so do not activate it unless required.

SENSe<Sensor>:[POWER:][AVG:]SMOothing:STATe <state>

Enables or disables the smoothing filter, a steep-edge digital lowpass filter. If you cannot adjust the aperture time exactly to the modulation period, the filter reduces result fluctuations caused by modulation.

Parameters:

<state> ON | OFF
 *RST: OFF

Example: SMO:STAT OFF

8.6.1.3 Measuring modulated signals

When measuring modulated signals in continuous average mode, the measurement can show fluctuation due to the modulation. If that is the case, adapt the size of the sampling window exactly to the modulation period to get an optimally stable display. If the modulation period varies or is not precisely known, you can also activate the smoothing function.

With smoothing activated, the selected sampling window has to be 5 to 9 times larger than the modulation period so that the fluctuations caused by modulation are sufficiently reduced. The sampling values are subjected to weighting (raised-von-Hann window), which corresponds to video filtering.

If you deactivate the smoothing filter, 300 to 3000 periods are required to obtain the same effect. The sampling values are considered equivalent and are averaged in a sampling window, which yields an integrating behavior of the measuring instrument. To obtain optimum suppression of variations in the result, exactly adapt the modulation

period to the size of the sampling window. Otherwise, the modulation can have a considerable influence, even if the sampling window is much larger than the modulation period.

8.6.1.4 Calculating the measurement time

Normally, the measurement time is calculated as follows:

$$MT = 2 * AC * APER + (2 * AC - 1) * 5 \mu s$$

With:

MT: overall measurement time

AC: average count

APER: aperture time

5 μ s is the time for switching the chopper phase.

8.6.1.5 Configuring the result buffer

<code>SENSe<Sensor>:[POWer:][AVG:]BUFFer:CLEar</code>	58
<code>SENSe<Sensor>:[POWer:][AVG:]BUFFer:COUNT?</code>	58
<code>SENSe<Sensor>:[POWer:][AVG:]BUFFer:SIZE</code>	58
<code>SENSe<Sensor>:[POWer:][AVG:]BUFFer:STATE</code>	59

SENSe<Sensor>:[POWer:][AVG:]BUFFer:CLEar

Clears the contents of the result buffer.

Example: `BUFF:CLE`

Usage: Event

SENSe<Sensor>:[POWer:][AVG:]BUFFer:COUNT?

Queries the number of results that are currently stored in the result buffer.

Example: `BUFF:COUN?`

Usage: Query only

SENSe<Sensor>:[POWer:][AVG:]BUFFer:SIZE <count>

Sets the size of the result buffer.

You can enable the buffer using `SENSe<Sensor>:[POWer:] [AVG:]BUFFer:STATE`

.

Parameters:

<count>	Range:	1 to 8192
	*RST:	1

Example: `BUFF:SIZE 1`
 See [Chapter 9.2, "Performing a buffered continuous average measurement"](#), on page 121.

SENSe<Sensor>:[POWer:][AVG:]BUFFer:STATe <state>

Enables or disables a buffered continuous average measurement. If enabled, the power sensor collects all results generated by trigger events until the buffer is filled.

You can set the size of the buffer using `SENSe<Sensor>:[POWer:] [AVG:]BUFFer:SIZE`.

Parameters:

<state> ON | OFF
 *RST: OFF

Example: `BUFF:STAT OFF`

8.6.2 Burst average measurement

The burst average mode is used to measure the average power of bursts. The integration time of a measurement is not predefined but determined by the power sensor with the aid of a burst detector. The start of a burst is detected when the measurement signal rises above a set trigger level. The measurement ends when the signal drops below a trigger threshold.

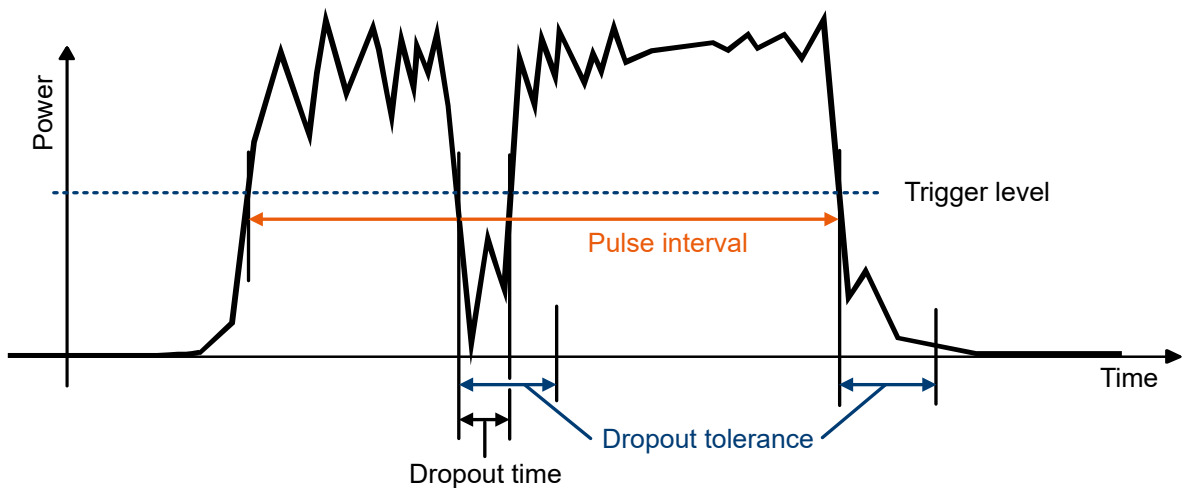


Figure 8-2: Burst average parameters



For information on querying the measurement results, see [Chapter 8.9.2, "Burst average measurement results"](#), on page 97.

8.6.2.1 Defining the dropout tolerance

To prevent power drops due to modulation from being erroneously interpreted as the end of a pulse, you must define the dropout tolerance. The dropout tolerance is a time interval in which the pulse end is only recognized if the signal level no longer exceeds the trigger level.

SENSe<Sensor>:[POWer:]BURSt:DTOLerance <tolerance>

Sets the dropout tolerance, a time interval in which the pulse end is only recognized if the signal level no longer exceeds the trigger level. See [Figure 8-2](#).

Parameters:

<tolerance>	Range:	0.00 to 0.003
	*RST:	1.000e-6
	Default unit:	Seconds

8.6.2.2 Defining a time interval for the measurement

At the beginning and at the end of the measurement interval, you can define time intervals that are excluded from the measurement, see [Chapter 8.7.3, "Excluding intervals"](#), on page 78.

8.6.2.3 Triggering a burst average measurement

In burst average mode, only internal trigger events from the signal are evaluated, irrespective of the setting of the [TRIGger:SOURce](#) parameter. The [TRIGger:DELay](#) parameter is also ignored, so that the measurement interval begins exactly when the signal exceeds the trigger level.

8.6.2.4 Querying the pulse interval

SENSe<Sensor>:[POWer:]BURSt:LENGth?

Queries the length of a burst (pulse interval), the time between the trigger point of the measurement and the time the trigger logic detects the end of the pulse. See [Figure 8-2](#).

Usage: Query only

8.6.3 Timeslot measurement

The timeslot mode is used to measure the average power of a definable number of successive timeslots within a frame structure with equal spacing. The measurement result is an array with the same number of elements as timeslots. Each element represents the average power in a particular timeslot.

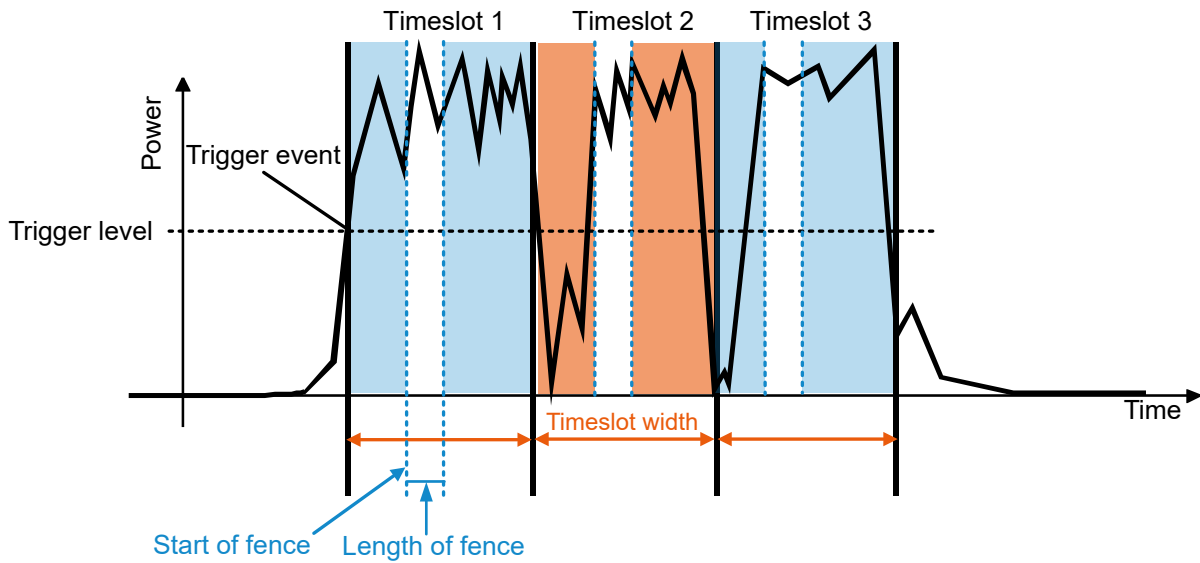


Figure 8-3: Timeslot parameters



For information on querying the measurement results, see [Chapter 8.9.3, "Timeslot measurement results"](#), on page 97.

8.6.3.1 Triggering a timeslot measurement

In timeslot mode, internal and external trigger events from the signal are evaluated depending on the settings of the `TRIGGER:SOURCE` parameter. It is essential to define the `TRIGGER:DELAY` parameter to ensure that the beginning of the first slot to be measured coincides with the delayed trigger point.

8.6.3.2 Defining a time interval for the measurement

At the beginning and at the end of the measurement interval, you can define time intervals that are excluded from the measurement, see [Chapter 8.7.3, "Excluding intervals"](#), on page 78.

8.6.3.3 Configuring the time slots

<code>SENSe<Sensor>:[POWER:]TSLot[:AVG]:COUNT</code>	61
<code>SENSe<Sensor>:[POWER:]TSLot[:AVG]:WIDTH</code>	62
<code>SENSe<Sensor>:[POWER:]TSLot[:AVG][:EXCLUDE]:MID:OFFSet[:TIME]</code>	62
<code>SENSe<Sensor>:[POWER:]TSLot[:AVG][:EXCLUDE]:MID:TIME</code>	62
<code>SENSe<Sensor>:[POWER:]TSLot[:AVG][:EXCLUDE]:MID[:STATE]</code>	62

`SENSe<Sensor>:[POWER:]TSLot[:AVG]:COUNT <count>`

Sets the number of simultaneously measured timeslots. See [Figure 8-3](#).

Parameters:

<count> Range: 1 to 16
 *RST: 8

SENSe<Sensor>:[POWER:]TSLot[:AVG]:WIDTh <width>

Sets the length of the timeslot. See [Figure 8-3](#).

Parameters:

<width> Range: 50.0e-9 to 0.10
 *RST: 1.000e-3
 Default unit: Seconds

SENSe<Sensor>:[POWER:]TSLot[:AVG][:EXCLude]:MID:OFFSet[:TIME] <time>

Determines the distance from the start of the timeslots to the start of the interval to be blanked out. See [Figure 8-3](#).

Parameters:

<time> Range: 0.00 to 0.10
 *RST: 0.00
 Default unit: Seconds

SENSe<Sensor>:[POWER:]TSLot[:AVG][:EXCLude]:MID:TIME <time>

Sets the length of the time interval in the timeslots to be excluded from the measurement. See [Figure 8-3](#). The parameter applies to each individual timeslot.

Note: Even if the exclusion interval exceeds the timeslot because, for example, its right limit is outside the timeslot, correct results are obtained. In the extreme case, where the interval length has been set to a value greater than the timeslot length, 0 W is output as the measured power. No error message is output.

Parameters:

<time> Range: 0.00 to 0.10
 *RST: 0.00
 Default unit: Seconds

SENSe<Sensor>:[POWER:]TSLot[:AVG][:EXCLude]:MID[:STATe] <state>

Enables or disables the blanking out of time intervals in the timeslots.

Parameters:

<state> *RST: OFF

8.6.4 Trace measurement

The trace measurement determines the course of power over a defined time. During the measurement time set by **SENSe<Sensor>:TRACe:TIME**, a large number of mea-

measurements are performed. The result is returned as an array of values with a size predefined by `SENSe<Sensor>:TRACe:POINts`. The length of an individual measurement(-point) is determined from the ratio of measurement time and measurement points. The entire result is called a "trace". Each trace must be triggered separately.

In trace mode, you can also perform automated pulse analysis. See [Chapter 8.6.5, "Pulse analysis measurement"](#), on page 67.



For information on querying the measurement results, see [Chapter 8.9.4, "Trace measurement results"](#), on page 97.

8.6.4.1 Configuring the video bandwidth

The number of analyzed samples is the product of the analysis window length, the number of repetitions and the sampling rate. In turn, the sampling rate is a function of the video bandwidth.

<code>SENSe<Sensor>:BWIDth:VIDeo</code>	63
<code>SENSe<Sensor>:BWIDth:VIDeo:LIST?</code>	64

`SENSe<Sensor>:BWIDth:VIDeo <value>`

Available in trace and statistics modes.

Reduces the video bandwidth, thus increasing the trigger sensitivity and reducing the display noise. To prevent signals from being corrupted, do not select a video bandwidth smaller than the RF bandwidth of the measurement signal.

If you reduce the video bandwidth, the sampling rate is also automatically reduced. In trace mode, the effective time resolution is reduced accordingly. In statistics mode, you have to increase the measurement time if you want to maintain the sample size.

Table 8-2: Effect on sampling rate and sampling interval

Video bandwidth	Sampling rate	Sampling interval
"Full"	$8 \times 10^7 \text{ s}^{-1}$	12.5 ns
"5 MHz"	$4 \times 10^7 \text{ s}^{-1}$	25 ns
"1.5 MHz"	$1 \times 10^7 \text{ s}^{-1}$	100 ns
"300 kHz"	$2.5 \times 10^6 \text{ s}^{-1}$	400 ns

Parameters:

<value>

"FULL"

The effect depends on the frequency set by `SENSe<Sensor>:FREQuency`:

Frequency \geq 500 MHz: Video bandwidth of at least 30 MHz is set.

Frequency < 500 MHz: Video bandwidth of approx. 7.5 MHz is set.

"5 MHZ"
5 MHz
"1.5 MHZ"
1.5 MHz
"300 KHZ"
300 kHz
*RST: "FULL"

SENSe<Sensor>:BWIDth:VIDeo:LIST?

Available in trace and statistics modes.

Lists all available video bandwidth settings.

Usage: Query only

8.6.4.2 Configuring the trace measurement

SENSe<Sensor>:TRACe:AVERage:COUNT.....	64
SENSe<Sensor>:TRACe:AVERage:TCONtrol.....	64
SENSe<Sensor>:TRACe:AVERage[:STATe].....	65
SENSe<Sensor>:TRACe:FFT:WINDow.....	65
SENSe<Sensor>:TRACe:MPWidth?.....	66
SENSe<Sensor>:TRACe:OFFSet:TIME.....	66
SENSe<Sensor>:TRACe:POINts.....	66
SENSe<Sensor>:TRACe:POINts:FPGA?.....	66
SENSe<Sensor>:TRACe:TIME.....	66
SENSe<Sensor>:TRACe:TIME:FPGA?.....	67

SENSe<Sensor>:TRACe:AVERage:COUNT <count>

Available in trace and statistics modes.

Sets the number of readings that are averaged for one measured value. The higher the count, the lower the noise, and the longer it takes to obtain a measured value.

Averaging is only effective, if `SENSe<Sensor>:TRACe:AVERage[:STATe] ON` is set.

Parameters:

<count> Range: 1 to 65536
*RST: 4

SENSe<Sensor>:TRACe:AVERage:TCONtrol <mode>

Defines how the measurement results are output. This is called termination control.

See also [Chapter 8.5, "Controlling the measurement"](#), on page 42.

Parameters:

<mode> MOVing | REPeat

MOVing

Outputs intermediate values to facilitate early detection of changes in the measured quantity. In the settled state, that means when the number of measurements specified by the average count has been performed, a moving average is output.

REPeat

Specifies that a measurement result is not output until the entire measurement has been completed. This means that the number of measurement cycle repetitions is equal to the set average count. If the average count is large, the measurement time can be very long.

The average count is set using `SENSe<Sensor>:TRACe: AVERage:COUNT`.

*RST: REPeat

Example:

TRAC: AVER: TCON REP

SENSe<Sensor>:TRACe:AVERage[:STATE] <state>

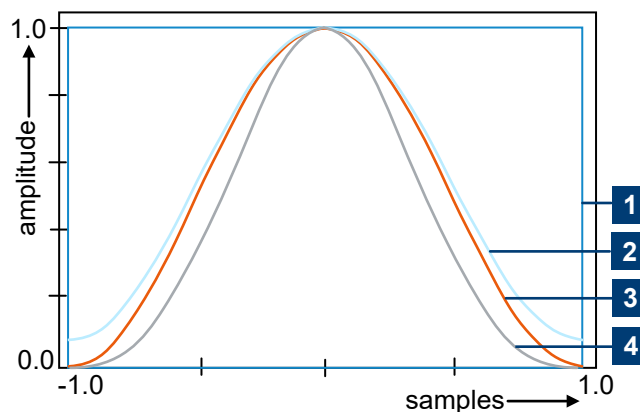
Enables or disables the averaging filter.

Parameters:

<state> *RST: ON

SENSe<Sensor>:TRACe:FFT:WINDow <window>

Selects a window function for computing a fast Fourier transform (FFT) of the trace. The resulting power spectrum of the trace is queried by `SENSe<Sensor>:TRACe: FFT: DATA?`.



- 1 = rectangular window
- 2 = Hamming window
- 3 = Hanning window
- 4 = Blackman window

Parameters:

<window> RECTangular | HAMMING | HANNing | BLACKman
 *RST: RECTangular

SENSe<Sensor>:TRACe:MPWidth?

Queries the attainable time resolution. The result is the smallest possible distance between two pixels, i.e. it is the smallest time interval that can be assigned to a pixel.

Usage: Query only

SENSe<Sensor>:TRACe:OFFSet:TIME <time>

Adds an offset to the beginning of the trace sequence. Thus, the trace in the result display is moved in positive or negative x-direction. If you measure with more than one power sensor, you can use this offset to arrange the traces to each other. The start of recording relative to the trigger event is set using [TRIGger:DElay](#).

Parameters:

<time> Range: Depends on the trigger delay.
 *RST: 0.0
 Default unit: Seconds

Example: TRAC:OFFS:TIME 1.0

SENSe<Sensor>:TRACe:POINts <points>

Sets the number of required values per trace sequence.

Parameters:

<points> Range: 1 to 8192
 *RST: 200

SENSe<Sensor>:TRACe:POINts:FPGA?

Queries the number of points used internally by the signal processing FPGA.

Usage: Query only

SENSe<Sensor>:TRACe:TIME <time>

Sets the trace length, time to be covered by the trace sequence. This time period is divided into several equal intervals, in which the average power is determined. The number of intervals equals the number of trace points, which is set using [SENSe<Sensor>:TRACe:POINts](#).

Parameters:

<time> Range: 10.0e-6 to 3.0
 *RST: 0.01
 Default unit: Seconds

SENSe<Sensor>:TRACe:TIME:FPGA?

Queries the trace length used internally by the signal processing FPGA.

Usage: Query only

8.6.5 Pulse analysis measurement

Use this measurement for automatic analysis of pulsed signals. You can measure either power over time over the whole trace, or restrict the measurement to a defined portion of the pulse signal. Trigger each trace separately.

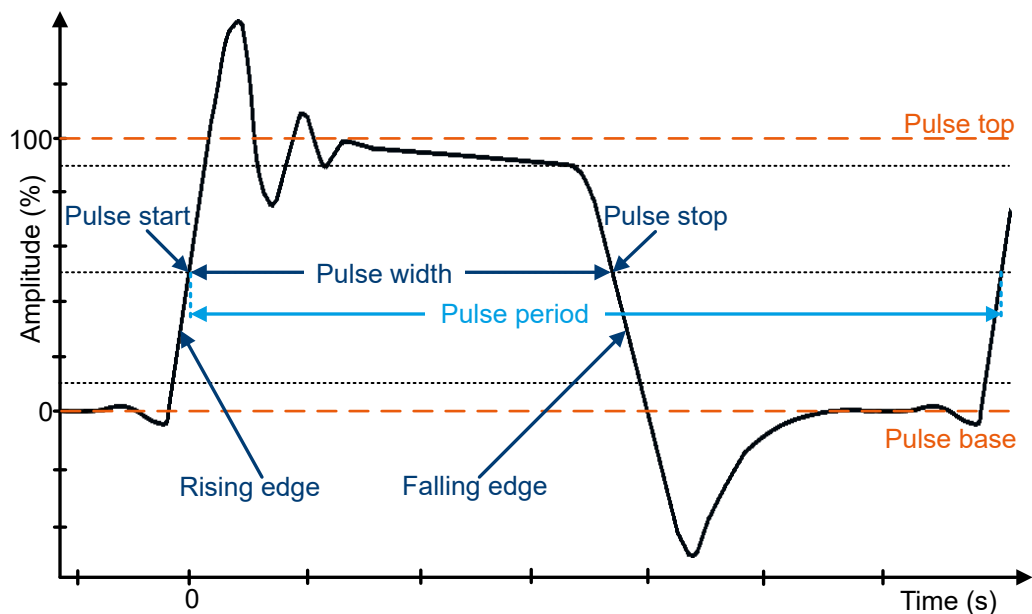


Figure 8-4: Pulse characteristics



For information on querying the measurement results, see [Chapter 8.9.5, "Pulse analysis measurement results"](#), on page 100.

8.6.5.1 Enabling the pulse analysis measurement

Pulse analysis measurements are performed in trace mode. Select the trace mode using `SENSe<Sensor>:FUNction`.

SENSe<Sensor>:TRACe:MEASurement:STATe <state>

Available in trace mode.

Enables or disables the pulse analysis measurement.

Parameters:

<state> ON | OFF

ON

The power sensor automatically determines the pulse parameters for the currently measured trace.

OFF

Pulse analysis measurement disabled.

*RST: OFF

8.6.5.2 Setting the duration

SENSe<Sensor>:TRACe:MEASurement:TIME <time>

Available in pulse analysis measurements.

Sets the duration of the pulse analysis. The start time of the pulse analysis is set by [SENSe<Sensor>:TRACe:MEASurement:OFFSet:TIME](#). The end time of the pulse analysis is set by this command in relation to the end of the trace measurement.

Parameters:

<time> A value of 0.0 means that the pulse analysis ends with the end of the trace measurement.
A value larger than 0.0 shortens the duration of the pulse analysis accordingly.

Range: 0.0 to 10.0
*RST: 0.0
Default unit: s

8.6.5.3 Defining reference levels for pulse analysis measurements

Define the reference levels for the pulse timing. All values are specified in percent of the pulse amplitude.

Remote commands:

SENSe<Sensor>:TRACe:MEASurement:ALGorithm	68
SENSe<Sensor>:TRACe:MEASurement:DEFine:DURation:REFerence	69
SENSe<Sensor>:TRACe:MEASurement:DEFine:TRANsition:HREFerence	69
SENSe<Sensor>:TRACe:MEASurement:DEFine:TRANsition:LREFerence	69

SENSe<Sensor>:TRACe:MEASurement:ALGorithm <algorithm>

Sets the analysis algorithm for detecting the pulse top and the pulse base power of a pulsed signal. From these two power levels, the power sensor derives the reference levels, using the probability density of the recorded trace. If the quality of the signal is too low to identify the pulse top and the pulse base power, the power sensor uses the maximum and minimum values of the trace instead.

Parameters:

<algorithm> HISTogram | INTegration | PEAK

HISTogram

Analyzes the histogram of the trace data. The average of all points representing the pulse top is taken as pulse top power. Similarly, the pulse base is determined.

This algorithm is suitable for most pulse signals.

INTEgration

Fits a rectangle pulse of the same energy into the pulse signal as a reference and thus determines the pulse top power.

This algorithm is recommended for:

Pulse signals with modulation

If the pulse energy is considered; for example, if you compare the measurement result to a measurement result of a thermal power sensor.

PEAK

Assigns the pulse peak power to the pulse top power, and the pulse minimum power to the pulse base power.

Suitable for distance-measuring equipment (DME) pulses.

*RST: HISTogram

SENSe<Sensor>:TRACe:MEASurement:DEFine:DURation:REFerence

<reference>

Sets the pulse width, pulse start time and pulse stop time. Used for the measurement of pulse duration and period.

Parameters:

<reference> Specified in percent of the pulse amplitude.
 Range: 0.0 to 100.0
 *RST: 50.0
 Default unit: PCT

SENSe<Sensor>:TRACe:MEASurement:DEFine:TRANSition:HREFerence

<hreference>

Sets the end of the rising edge and the start of the falling edge of the pulse. Used for the measurement of the rise or fall time.

Parameters:

<hreference> Specified in percent of the pulse amplitude.
 Range: 0.0 to 100.0
 *RST: 90.0
 Default unit: PCT

SENSe<Sensor>:TRACe:MEASurement:DEFine:TRANSition:LREFerence

<lreference>

Sets the start of the rising edge and the start of the falling edge of the pulse. Used for the measurement of the rise or fall time.

Parameters:

<Ireference> Specified in percent of the pulse amplitude.
 Range: 0.0 to 100.0
 *RST: 10.0
 Default unit: PCT

8.6.5.4 High-resolution measurements

`SENSe<Sensor>:TRACe:ESAMpling:AUTO`..... 70

`SENSe<Sensor>:TRACe:MEASurement:TRANSition:ESAMpling:AUTO`..... 70

SENSe<Sensor>:TRACe:ESAMpling:AUTO <value>

Available in trace mode.

Enables or disables the automatic equivalent sampling that allows for high-resolution measurements.

The automatic equivalent sampling requires a periodic measurement signal. The power sensor superposes several successive measurements to improve the resolution. The measurement speed decreases, because several successive measurements are necessary.

Parameters:

<value> ON | OFF
 *RST: ON

SENSe<Sensor>:TRACe:MEASurement:TRANSition:ESAMpling:AUTO <val>

Available in pulse analysis measurements.

Enables or disables the automatic equivalent sampling that allows for high-resolution measurements.

The automatic equivalent sampling requires a periodic measurement signal. The power sensor superposes several successive measurements to improve the resolution. The measurement speed decreases, because several successive measurements are necessary.

Parameters:

<val> OFF | ON
 *RST: ON

8.6.6 Statistics measurement

Using the statistics mode, you can analyze the statistical distribution of the envelope power. The duration of the sampling window is either defined by the chosen gate or a set aperture time. The measurement is repeated until the minimum number of samples is collected.

Two functions for analyzing the statistical distribution of the envelope power are available.

Complementary cumulative distribution function (CCDF)

Probability that the envelope power is higher than the corresponding x-axis power value. Linear or logarithmic scale.

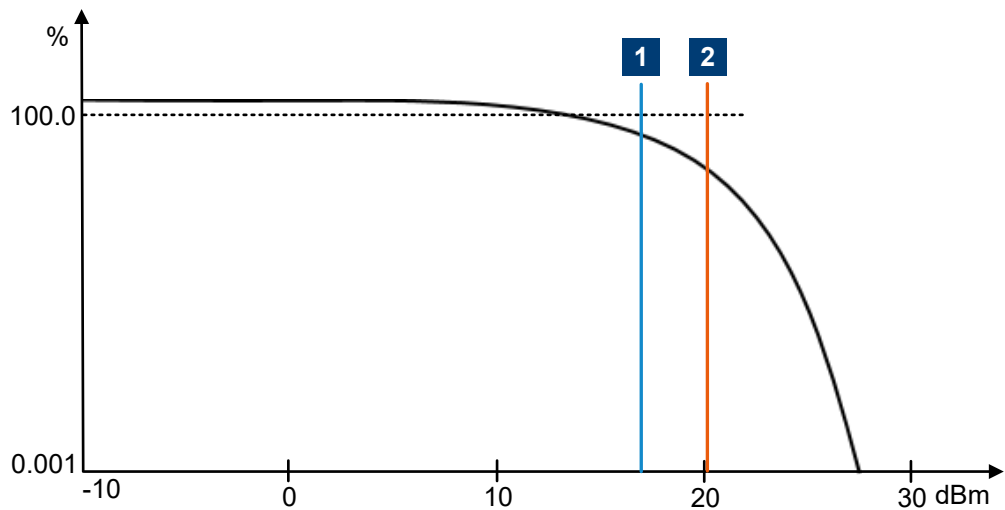


Figure 8-5: CCDF measurement parameters

1 = Average power
2 = Peak power

Probability density function (PDF)

Normalized distribution density of the envelope power. The measurement results are dimensionless and independent of the magnitude of the average power value (A_v). Only linear scale is available.

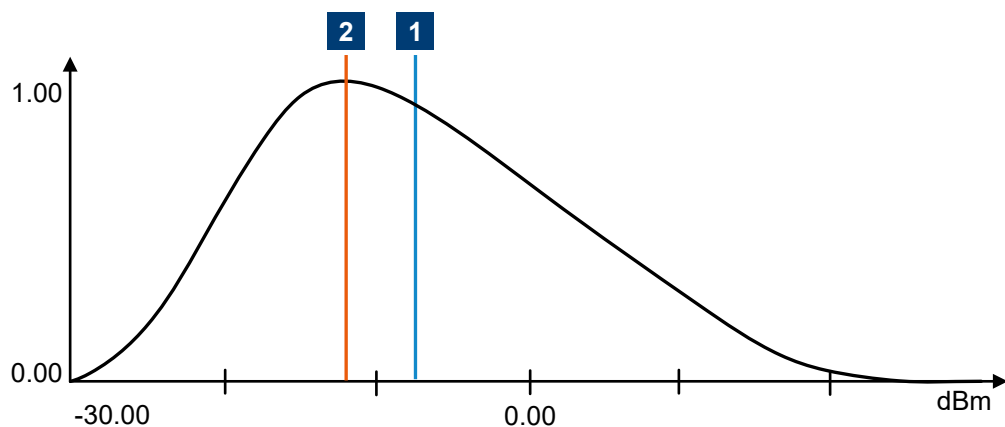


Figure 8-6: PDF measurement parameter

1 = Average power
2 = Peak power



For information on querying the measurement results, see [Chapter 8.9.6, "Statistics measurement results"](#), on page 107.

8.6.6.1 Configuring the analysis window

Set the number of analysis window repetitions using `SENSe<Sensor>:TRACe:AVERAge:COUNT`. For defining start, length and exclusion period, use the following commands:

<code>SENSe<Sensor>:STATistics[:EXCLude]:MID:OFFSet[:TIME]</code>	72
<code>SENSe<Sensor>:STATistics[:EXCLude]:MID:TIME</code>	72
<code>SENSe<Sensor>:STATistics:OFFSet:TIME</code>	72
<code>SENSe<Sensor>:STATistics:TIME</code>	73

SENSe<Sensor>:STATistics[:EXCLude]:MID:OFFSet[:TIME] <time>

Sets the start of an exclusion interval in the analysis window, referenced to the start of the analysis window.

All other settings are the same as for the timeslot mode, see [Chapter 8.6.3, "Timeslot measurement"](#), on page 60.

Parameters:

<time>	Range:	0.0 to 0.3
	*RST:	0.0
	Default unit:	s

SENSe<Sensor>:STATistics[:EXCLude]:MID:TIME <time>

Sets the length of an exclusion interval in the analysis window.

All other settings are the same as for the timeslot mode, see [Chapter 8.6.3, "Timeslot measurement"](#), on page 60.

Parameters:

<time>	Range:	0.0 to 0.3
	*RST:	0.0
	Default unit:	s

SENSe<Sensor>:STATistics:OFFSet:TIME <time>

Sets the start of the analysis window for recording measured values. The start time is referenced to the beginning of the actual measurement. If the analysis window starts before the trigger event, you have to set a sufficient negative trigger delay using `TRIGger:DELaY`, at least the minimum value plus a safety margin of $-51.2 \mu\text{s}$.

Parameters:

<time>	Range:	0.0 to 10.0
	*RST:	0.0
	Default unit:	s

SENSe<Sensor>:STATistics:TIME <time>

Sets the duration of the statistics measurement.

Parameters:

<time> Range: 50.0e-9 to 53
 *RST: 0.01
 Default unit: s

8.6.6.2 Configuring the video bandwidth

See [Chapter 8.6.4.1, "Configuring the video bandwidth"](#), on page 63.

8.6.6.3 Triggering a statistics measurement

You can trigger the statistical analysis by `INTERNAL`, `EXTERNAL` or `IMMEDIATE` trigger sources. See [Chapter 8.5.2.3, "Trigger sources"](#), on page 44.

8.6.6.4 Querying the highest level resolution**SENSe<Sensor>:STATistics:SCALE:X:MPWidth?**

Queries the highest level resolution that is attainable. The value is fixed at 0.006 dB per pixel. If this value is not reached, the power sensor outputs a settings conflict message.

Solutions for the conflict can be:

- Select a smaller number of pixels using `SENSe<Sensor>:STATistics:SCALE:X:POINTS`.
- Choose a larger width for the level range using `SENSe<Sensor>:STATistics:SCALE:X:RANGE`.

Usage: Query only

8.6.6.5 Configuring the scale

Set the level range and its resolution in pixels for the measurement result. For each pixel, the selected function, CCDF or PDF, is output in W^{-1} .

<code>SENSe<Sensor>:STATistics:SCALE:X:POINTS</code>	73
<code>SENSe<Sensor>:STATistics:SCALE:X:RANGE</code>	74
<code>SENSe<Sensor>:STATistics:SCALE:X:RLEVEL</code>	74

SENSe<Sensor>:STATistics:SCALE:X:POINTS <points>

Sets the resolution of the measurement result. The defined number of pixels is assigned to the logarithmic level range, set by `SENSe<Sensor>:STATistics:SCALE:X:RANGE`.

Parameters:

<points> Range: 3 to 8191
 *RST: 200

SENSe<Sensor>:STATistics:SCALE:X:RANGe <range>

Sets the level range width of the measurement result.

The level range width divided by N-1, where N is the number of pixels, has to equal at least the highest attainable level resolution, queried by [SENSe<Sensor>:STATistics:SCALE:X:MPWidth?](#). N is the number of pixels, set by [SENSe<Sensor>:STATistics:SCALE:X:POINTs](#).

Parameters:

<range> Range: 0.01 to 100
 *RST: 50
 Default unit: dB

SENSe<Sensor>:STATistics:SCALE:X:RLEVel <rlevel>

Sets the lower limit for the level range of the measurement result.

This level can be assigned to the first pixel. The level assigned to the last pixel is equal to the level of the first pixel plus the level range.

Parameters:

<rlevel> Range: Depends on the level range.
 *RST: -30.0
 Default unit: dBm

8.7 Configuring basic measurement parameters

The following section describes the settings common for several measurement modes.

8.7.1 Configuring auto averaging

Describes the commands for automatic averaging in continuous average and burst average measurements.

Remote commands:

SENSe<Sensor>:AVERage:COUNT	75
SENSe<Sensor>:AVERage:COUNT:AUTO	75
SENSe<Sensor>:AVERage:COUNT:AUTO:MTIME	75
SENSe<Sensor>:AVERage:COUNT:AUTO:NSRatio	76
SENSe<Sensor>:AVERage:COUNT:AUTO:RESolution	76
SENSe<Sensor>:AVERage:COUNT:AUTO:SLOT	76
SENSe<Sensor>:AVERage:RESet	77

SENSe<Sensor>:AVERAge:COUNT:AUTO:TYPE	77
SENSe<Sensor>:AVERAge:TCONtrol	77
SENSe<Sensor>:AVERAge[:STATe]	78

SENSe<Sensor>:AVERAge:COUNT <count>

Sets the number of readings that are averaged for one measured value. The higher the count, the lower the noise, and the longer it takes to obtain a measured value.

Average count is often also called averaging factor, but it designates the same thing, i.e. the number of measured values that have to be averaged for forming the measurement result.

Averaging is only effective, if [SENSe<Sensor>:AVERAge\[:STATe\]](#) ON is set.

Parameters:

<count>	Range:	1 to 1048576
	*RST:	1024

Example: AVER: COUN 1

SENSe<Sensor>:AVERAge:COUNT:AUTO <state>

Sets how the average count is determined.

Parameters:

<state>	ON
	Auto averaging: the averaging factor is continuously determined and set depending on the power level and other parameters.
	OFF
	Fixed filter: the previous, automatically determined averaging factor is used.
	ONCE
	Automatically determines an averaging factor under the current measurement conditions. Then changes to the fixed filter setting (see OFF) and uses this averaging factor.
	*RST: ON

SENSe<Sensor>:AVERAge:COUNT:AUTO:MTIME <maximum_time>

Sets an upper limit for the settling time of the auto-averaging filter if [SENSe<Sensor>:AVERAge:COUNT:AUTO:TYPE](#) is set to `NSRatio`. Thus it limits the length of the filter.

Parameters:

<maximum_time>	Range:	0.01 to 999.99
	*RST:	4.00
	Default unit:	Seconds

SENSe<Sensor>:AVERAge:COUNT:AUTO:NSRatio <nsr>

Determines the relative noise component in the measurement result for the measurement modes with scalar results. These measurement modes are continuous average, burst average and timeslot, provided the particular power sensor supports them.

This command is only effective if the auto average calculation is enabled:

- `SENSe<Sensor>:AVERAge:COUNT:AUTO ON`
- `SENSe<Sensor>:AVERAge:COUNT:AUTO:TYPE NSR`

The noise component is defined as the magnitude of the level variation in dB caused by the inherent noise of the power sensor (two standard deviations).

The query returns the relative noise component in the measured value.

Parameters:

<nsr>	Range:	100.000e-6 to 1.00
	*RST:	0.01
	Default unit:	dB

SENSe<Sensor>:AVERAge:COUNT:AUTO:RESolution <resolution>

Defines the number of significant places for linear units and the number of decimal places for logarithmic units that are likely free of noise in the measurement result.

The setting is only considered, if the following applies:

- `SENSe<Sensor>:AVERAge:COUNT:AUTO ON`
- `SENSe<Sensor>:AVERAge:COUNT:AUTO:TYPE RES`

Parameters:

<resolution>	Range:	1 to 4
	*RST:	3

SENSe<Sensor>:AVERAge:COUNT:AUTO:SLOT <slot>

Available only in timeslot mode.

Sets a timeslot from which the measured value is used to determine the filter length automatically. The timeslot number must not exceed the number of the currently set timeslots.

Parameters:

<slot>	Range:	1 to 16
	*RST:	1

SENSe<Sensor>:AVERage:RESet

Deletes all previous measurement results that the averaging filter contains and initializes the averaging filter. The filter length gradually increases from 1 to the set averaging factor. Thus, trends in the measurement result become quickly apparent. Note that the measurement time required for the averaging filter to settle completely remains unchanged.

Use this command if:

- High averaging factor is set.
`SENSe<Sensor>:AVERage:COUNT`
- Intermediate values are output as measurement results.
`SENSe<Sensor>:AVERage:TCONtrol MOVing`
- Power has significantly decreased since the previous measurement, for example by several powers of 10.

In this situation, previous measurement results, which are still contained in the averaging filter, strongly affect the settling of the display. As a result, the advantage of detecting trends in the measurement result while the measurement is still in progress is lost.

Example: `AVER:RES`

Usage: Event

SENSe<Sensor>:AVERage:COUNT:AUTO:TYPE <type>

Sets the automatic averaging filter.

Parameters:

<type> RESolution | NSRatio

RESolution

Usually used.

NSRatio

Predefines the compliance to an exactly defined noise component. Enter this value using `SENSe<Sensor>:CORRection:OFFSet`.

*RST: RESolution

SENSe<Sensor>:AVERage:TCONtrol <mode>

Defines how the measurement results are output. This is called termination control.

See also [Chapter 8.5, "Controlling the measurement"](#), on page 42.

Parameters:

<mode> MOVing | REPeat

MOVing

Outputs intermediate values to facilitate early detection of changes in the measured quantity. In the settled state, that means when the number of measurements specified by the average count has been performed, a moving average is output.

REPeat

Specifies that a measurement result is not output until the entire measurement has been completed. This means that the number of measurement cycle repetitions is equal to the set average count. If the average count is large, the measurement time can be very long.

The average count is set using `SENSe<Sensor>:AVERage:COUNT` on page 75.

*RST: REPeat

Example: AVER:TCON REP

SENSe<Sensor>:AVERage[:STATe] <state>

Available in continuous average, burst average and timeslot modes.

Enables or disables the averaging filter.

Parameters:

<state> *RST: ON

8.7.2 Setting the frequency

The frequency of the signal to be measured is not automatically determined. For achieving better accuracy, the carrier frequency of the applied signal must be set.

SENSe<Sensor>:FREQUENCY <frequency>

Transfers the carrier frequency of the RF signal to be measured. This frequency is used for the frequency-response correction of the measurement result.

The center frequency is set for broadband signals, e.g. spread-spectrum signals, multi-carrier signals.

Parameters:

<frequency> Range: 50.0e6 to 50.0e9
*RST: 1.0e9
Default unit: Frequency

Example: FREQ 10000

8.7.3 Excluding intervals

In the burst average and timeslot modes, you can define a time interval at the beginning or at the end of an integration interval that is excluded from the measurement result. See in [Figure 8-7](#).

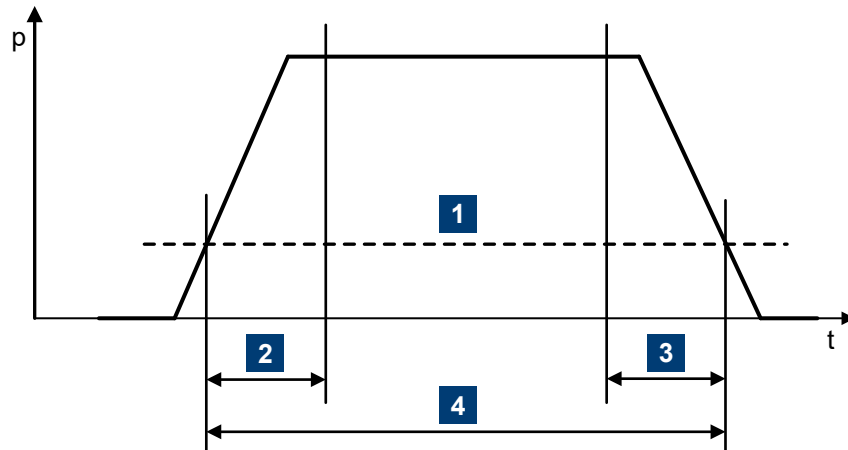


Figure 8-7: Excluding intervals

- 1 = Trigger level
- 2 = Excluding interval at the beginning
- 3 = Excluding interval at the end
- 4 = Integration interval

Remote commands:

<code>SENSe<Sensor>:TIMing:EXCLude:START</code>	79
<code>SENSe<Sensor>:TIMing:EXCLude:STOP</code>	79

SENSe<Sensor>:TIMing:EXCLude:START <exclude_start>

Available in burst average and timeslot modes.

Sets a time interval at the beginning of bursts that is excluded from the measurement. See [Figure 8-7](#).

Parameters:

<exclude_start> Range: 0.0 to 10.0
 *RST: 0.0
 Default unit: Seconds

SENSe<Sensor>:TIMing:EXCLude:STOP <exclude_stop>

Available in burst average and timeslot modes.

Sets a time interval at the end of bursts that is excluded from the measurement. See [Figure 8-7](#).

Parameters:

<exclude_stop> Range: 0.0 to 51.6e-6
 *RST: 0.0
 Default unit: Seconds

8.7.4 Configuring corrections

It is possible to set some parameters that compensate for a change of the measured signal due to fixed external influences.

- [Duty cycle corrections](#).....80
- [Offset corrections](#).....80
- [S-parameter correction](#)..... 81
- [S-gamma corrections](#).....82
- [Using the S-Parameters program](#)..... 84

8.7.4.1 Duty cycle corrections

The duty cycle is the percentage of one period during which the signal is active, when pulse-modulated signals are corrected. The duty cycle is only evaluated in the continuous average mode.

Remote commands:

- [SENSe<Sensor>:CORRection:DCYClE](#).....80
- [SENSe<Sensor>:CORRection:DCYClE:STATe](#).....80

SENSe<Sensor>:CORRection:DCYClE <duty_cycle>

Available in continuous average mode.

Sets the duty cycle for measuring pulse-modulated signals. The duty cycle defines the percentage of one period during which the signal is active. If the duty cycle is enabled, the R&S NRPxxP considers this percentage when calculating the signal pulse power from the average power.

Parameters:

<duty_cycle>	Range:	0.001 to 100.00
	*RST:	1.00
	Default unit:	Percent

SENSe<Sensor>:CORRection:DCYClE:STATe <state>

Enables or disables the duty cycle correction for the measured value.

Parameters:

<state>	*RST:	OFF
---------	-------	-----

8.7.4.2 Offset corrections

The offset accounts for external losses by adding a fixed level offset in dB.

The attenuation of an attenuator located ahead of the power sensor (or the coupling attenuation of a directional coupler) is considered with a positive offset. That means the power sensor calculates the power at the input of the attenuator or the directional coupler. A negative offset can be used to correct the influence of an amplifier connected ahead.

Using S-parameters instead of a fixed offset allows more precise measurements, because the interaction between the power sensor and the component can be considered. See also [Chapter 8.7.4.3, "S-parameter correction"](#), on page 81.

Remote commands:

SENSe<Sensor>:CORRection:OFFSet	81
SENSe<Sensor>:CORRection:OFFSet:STATe	81

SENSe<Sensor>:CORRection:OFFSet <offset>

Sets a fixed offset that is added to correct the measured value.

Parameters:

<offset>	Range:	-200.00 to 200.00
	*RST:	0
	Default unit:	dB

SENSe<Sensor>:CORRection:OFFSet:STATe <state>

Enables or disables the offset correction.

Parameters:

<state>	*RST:	OFF
---------	-------	-----

Example: CORR:OFFS:STAT ON

8.7.4.3 S-parameter correction

S-parameter correction compensates for the losses and reflections introduced by a component — such as an attenuator, directional coupler, or matching pad — that is attached to a power sensor. Using S-parameters instead of a fixed offset increases measurement accuracy by accounting for the interaction between the power sensor and the component. It shifts the reference plane of the power sensor from its RF connector to the input of the device that is being applied externally.

All R&S NRPxxP power sensors allow compensating for the influence of any 2-port device between the signal source and the power sensor input. As a result, the firmware can calculate the power that the signal source actually delivers. Examples of such 2-port devices include attenuators, matching pads, minimum-loss pads and waveguide adapters. One precondition for such compensation is that you provide a complete set of S-parameter data for the 2-port device in the frequency range required by the application.

Configuring the S-parameter correction

SENSe<Sensor>:CORRection:SPDevice:LIST?	82
SENSe<Sensor>:CORRection:SPDevice:SElect	82
SENSe<Sensor>:CORRection:SPDevice:STATe	82

SENSe<Sensor>:CORRection:SPDevice:LIST?

Queries the list of the S-parameter data sets that have been downloaded to the power sensor. Returns the consecutive number and mnemonic of each data set.

Suffix:

<undef> 1..n

Usage:

Query only

SENSe<Sensor>:CORRection:SPDevice:SElect <num>

Selects a downloaded data set for S-parameter correction.

See also [Chapter 8.7.4.3, "S-parameter correction"](#), on page 81.

Suffix:

<undef> 1..n

Parameters:

<num> Range: 1 to 1999
*RST: 1; can differ if a calibration set defines another value.

SENSe<Sensor>:CORRection:SPDevice:STATe <state>

Enables or disables the S-parameter correction. If activated, uses the S-parameter data set selected by [SENSe<Sensor>:CORRection:SPDevice:SElect](#).

See also [Chapter 8.7.4.3, "S-parameter correction"](#), on page 81.

Suffix:

<undef> 1..n

Parameters:

<state> *RST: OFF; can differ if a calibration set defines another value.

Example:

```
CORR:SPD:SEL 1
Selects an S-parameter correction data set.
CORR:SPD:STAT ON
Enables the S-parameter correction.
```

8.7.4.4 S-gamma corrections

Using the complex reflection coefficient, you can determine the power P delivered by the signal source with considerably greater accuracy.

The coefficient of the signal source Γ_{source} is defined by its magnitude and phase:

- [SENSe<Sensor>:SGAMma:MAGNitude](#)
- [SENSe<Sensor>:SGAMma:PHASe](#)

The complex reflection coefficient Γ_{sensor} of the power sensor, which is also required for the correction, is prestored in the calibration data memory for many frequencies.

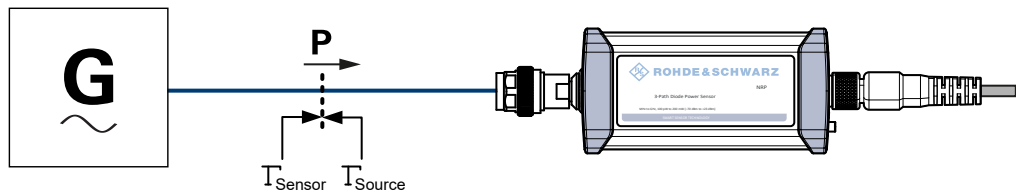


Figure 8-8: Correction of interactions between the power sensor and the signal source

If the gamma correction is performed in combination with an S-parameter correction (`SENSe<Sensor>:CORRection:SPDevice:STATe ON`), the following is considered:

- Interaction of the signal source with the S-parameter device
- Input of the power sensor, depending on the transmission, expressed by the term $S_{12}S_{21}$

The interaction between the complex reflection coefficient Γ_{sensor} of the power sensor and the reflection of port 2 is expressed by s_{22} , see Figure 8-9. If the S-parameter correction is enabled, this interaction is always considered, regardless whether gamma correction is performed or not.

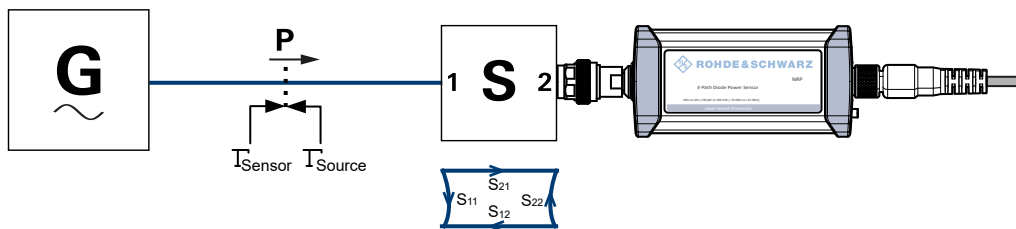


Figure 8-9: Correction of interactions between the power sensor, the signal source, and the S-parameter device

Remote commands:

<code>SENSe<Sensor>:SGAMma:CORRection:STATe</code>	83
<code>SENSe<Sensor>:SGAMma:MAGNitude</code>	83
<code>SENSe<Sensor>:SGAMma:PHASe</code>	84

`SENSe<Sensor>:SGAMma:CORRection:STATe <state>`

Enables or disables the use of the complex reflection coefficient to correct the interactions of power sensor and signal source.

Parameters:

<state> *RST: OFF

`SENSe<Sensor>:SGAMma:MAGNitude <magnitude>`

Sets the magnitude of the complex reflection coefficient of the source, Γ_{source} .

Parameters:

<magnitude> 0.0 \triangleq ideal matched source
 1.0 \triangleq total reflection
 Range: 0.0 to 1.0
 *RST: 0.0

SENSe<Sensor>:SGAMma:PHASe <phase>

Sets the phase angle of the complex reflection coefficient of the source, Γ_{source} .

Parameters:

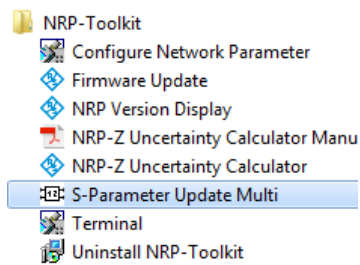
<phase> Range: -360.0 to 360.0
 *RST: 0.0

8.7.4.5 Using the S-Parameters program

The S-Parameters program helps loading an S-parameter table into the power sensor. The S-Parameters program is part of the R&S NRP Toolkit, see [Chapter 5.1, "R&S NRP Toolkit"](#), on page 22.

To start the S-Parameters program

- ▶ In the Windows start menu, select "NRP Toolkit" > "S-Parameter Update Multi".



User interface of the S-Parameters program

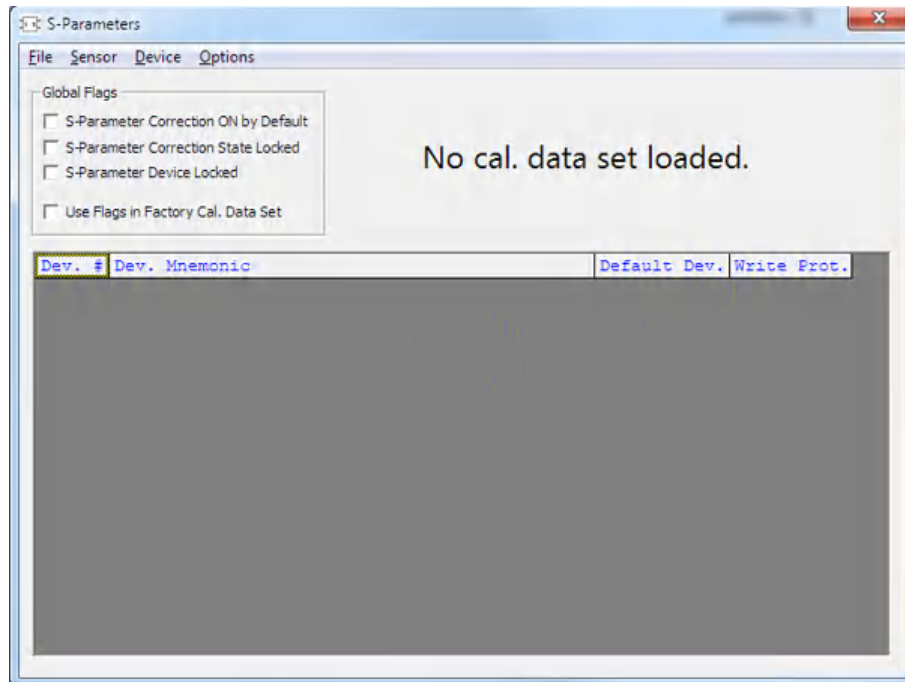


Figure 8-10: S-Parameters dialog

Menu bar.....	85
L File.....	85
L Sensor.....	86
L Device.....	86
L Options.....	86
L User Data.....	86
L Remote.....	86
L Show Cal. Data.....	86
Global Flags.....	87
L S-Parameter Correction ON by Default.....	87
L S-Parameter Correction State Locked.....	87
L S-Parameter Device Locked.....	87
L Use Flags in Factory Cal. Data Set.....	88
Device table.....	88

Menu bar

Contains the following submenus.

File ← Menu bar

Provides options for loading and saving calibration data files, see:

- "To change the S-parameter data" on page 90
- "To load an uncertainty parameter file" on page 91

Sensor ← Menu bar

Provides options for loading and saving calibration data directly from or to the power sensor, see:

- ["To load a calibration data set from a power sensor"](#) on page 88
- ["To save the calibration data on the power sensor"](#) on page 92

Device ← Menu bar

Provides functions for editing the table of S-parameter devices.

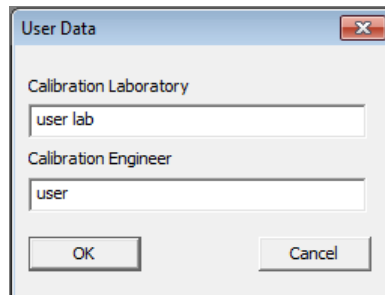
Options ← Menu bar

Provides functions for editing user data, changing remote control timeouts, and displaying calibration data as plain text.

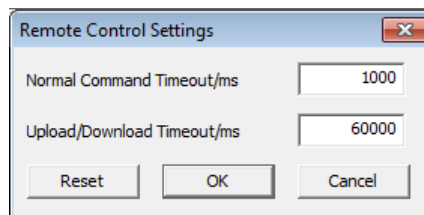
User Data ← Options ← Menu bar

Opens the "User Data" dialog.

Here, you can enter the name of the calibration laboratory and the calibration engineer that is stored in the calibration data set if changes are made.

**Remote ← Options ← Menu bar**

Opens the "Remote Control Settings" dialog. It is normally not necessary to change timeouts.

**Show Cal. Data ← Options ← Menu bar**

Displays the content of the calibration data set that has been loaded either from a file or directly from a power sensor as a plain text.

You can copy the text output to the clipboard by clicking "Copy to Clipboard."

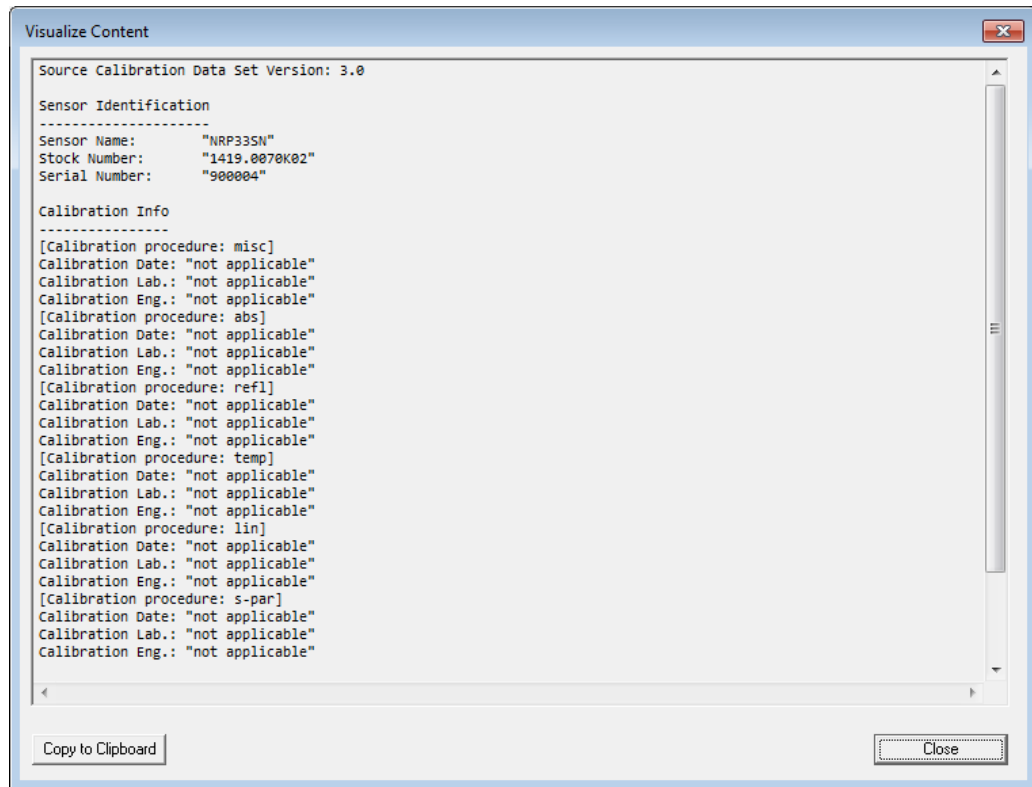


Figure 8-11: Example

Global Flags

Groups the settings for the power sensor behavior regarding S-parameter corrections.

S-Parameter Correction ON by Default ← Global Flags

If this option is enabled, the S-parameter correction is activated automatically when the power sensor is started.

S-Parameter Correction State Locked ← Global Flags

If enabled, the state that is selected with "S-Parameter Correction ON by Default" is locked and cannot be changed using:

- `SENSe<Sensor>:CORRection:SPDevice:STATE`
- R&S NRP2 base unit

S-Parameter Device Locked ← Global Flags

If enabled, the S-parameter device that is selected as the default device in the table of S-parameter devices is locked and cannot be changed using:

- `SENSe<Sensor>:CORRection:SPDevice:SElect`
- R&S NRP2 base unit

The default S-parameter device is the S-parameter device that you have selected when the power sensor is started.

Use Flags in Factory Cal. Data Set ← Global Flags

Available if the power sensor supports two different calibration data sets:

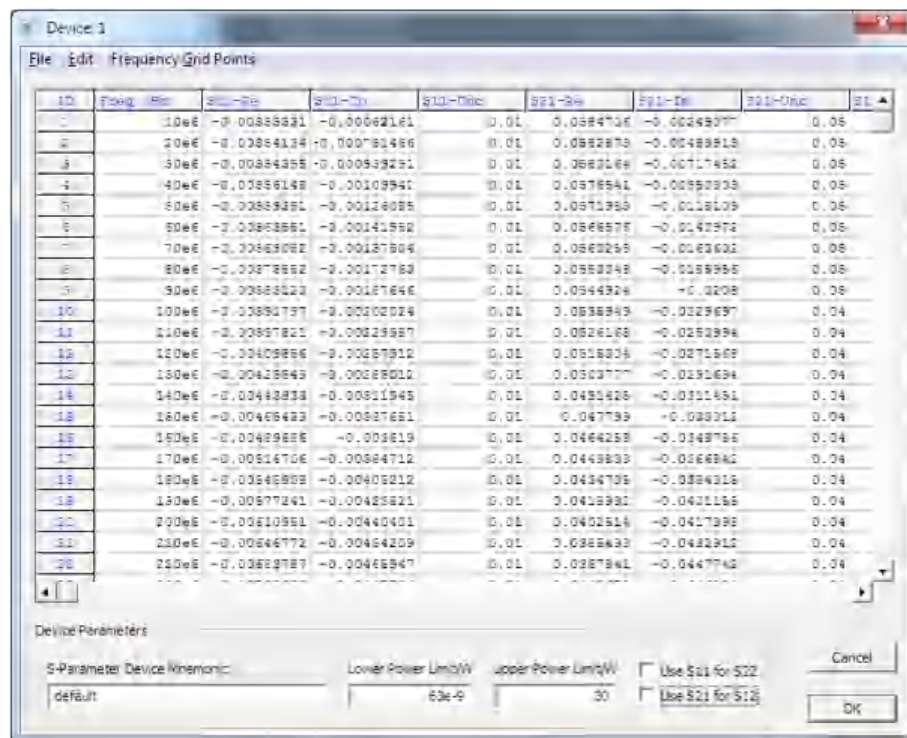
- Factory calibration data set containing all factory calibration data.
- User calibration data set that contains the S-parameter devices you have loaded.

Note: After you have added S-parameter devices and configured the global flags, disable this option. Otherwise, it is not possible to enable S-parameter correction because the flags in the factory calibration data set prevent it.

Device table

Shows a list of all S-parameter devices that are available in the calibration data set.

If you double-click an entry, a dialog for the device is opened that allows to import, export, and edit S-parameter data. See ["To change the S-parameter data"](#) on page 90.



Performing configuration tasks

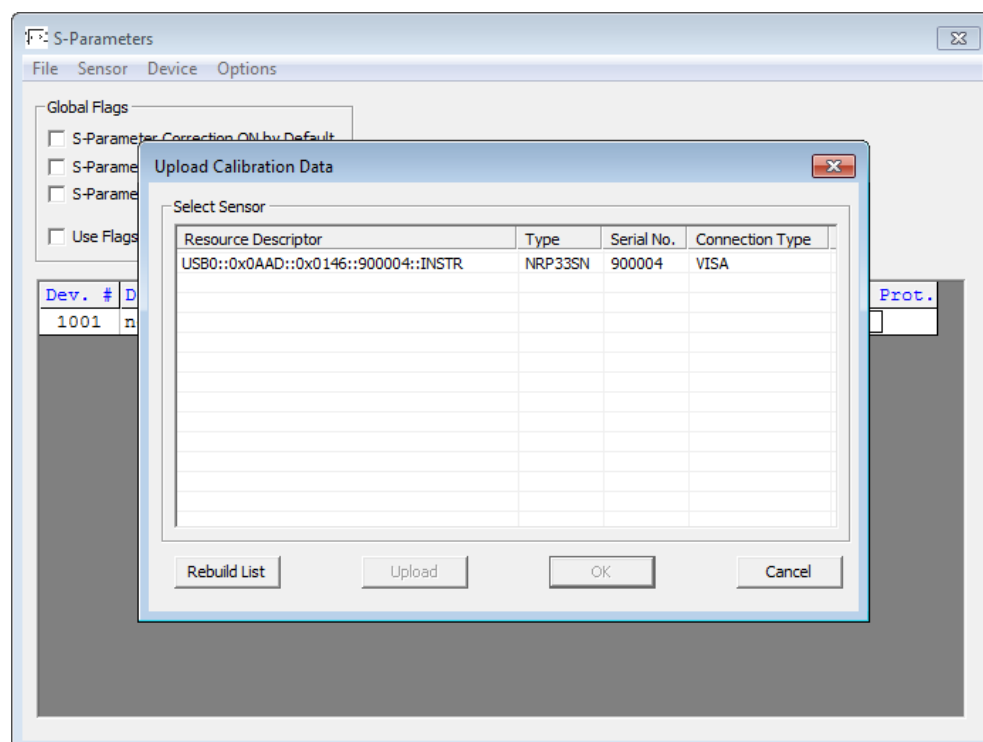
In this chapter, different configuration tasks performed with the power sensor and the "S-Parameter Update Multi" tool are described.

To load a calibration data set from a power sensor

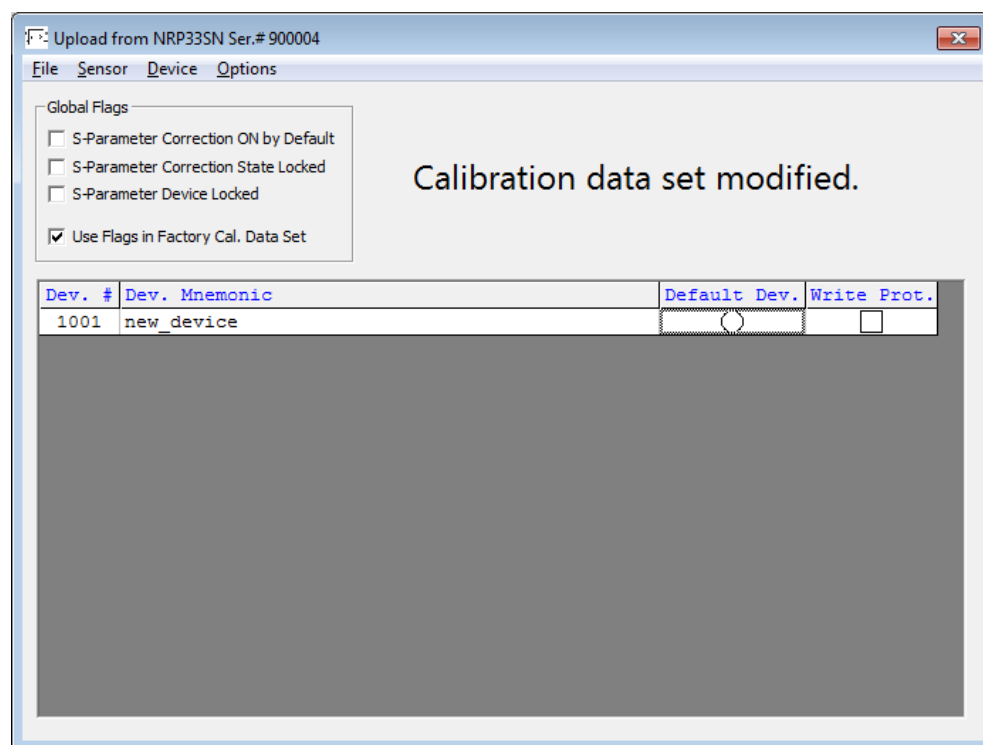
Prerequisites: The power sensor is connected to the computer and a connection is established.

1. Open the "S-Parameter Update Multi" program.
2. Select "Sensor" > "Load Calibration Data".

The "Upload Calibration Data" dialog opens. It shows a list of the available power sensors.



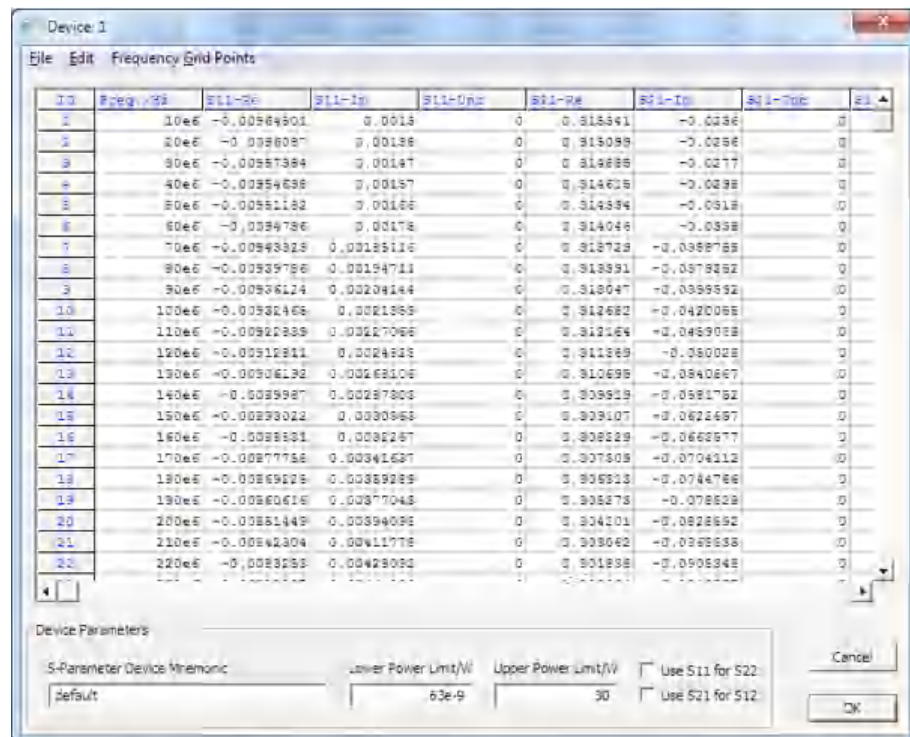
3. If you cannot find your power sensor in the list, for example because of reconnecting the power sensor, you can reload the list by clicking "Rebuild List".
4. Click "Upload" to load calibration data from the power sensor.
After the upload is finished, the "OK" button is enabled.
5. Click "OK" to apply the changes.
If you want to discard the changes, exit the dialog by clicking "Cancel".
After a successful upload, the name and serial number are shown in the name of the main dialog.



6. Create a backup of the calibration data set before making any changes.
Select "File" > "Save Calibration Data".
A dialog opens where you can select the location to save the calibration data.

To change the S-parameter data

1. In the device table, double-click an entry. See also "[Device table](#)" on page 88.
2. Select "File" > "Import S2P".
3. Select the *.S2P file you want to import.
4. Confirm with "Open".
The data from the selected file is loaded in the device table.
All uncertainties are set to zero.

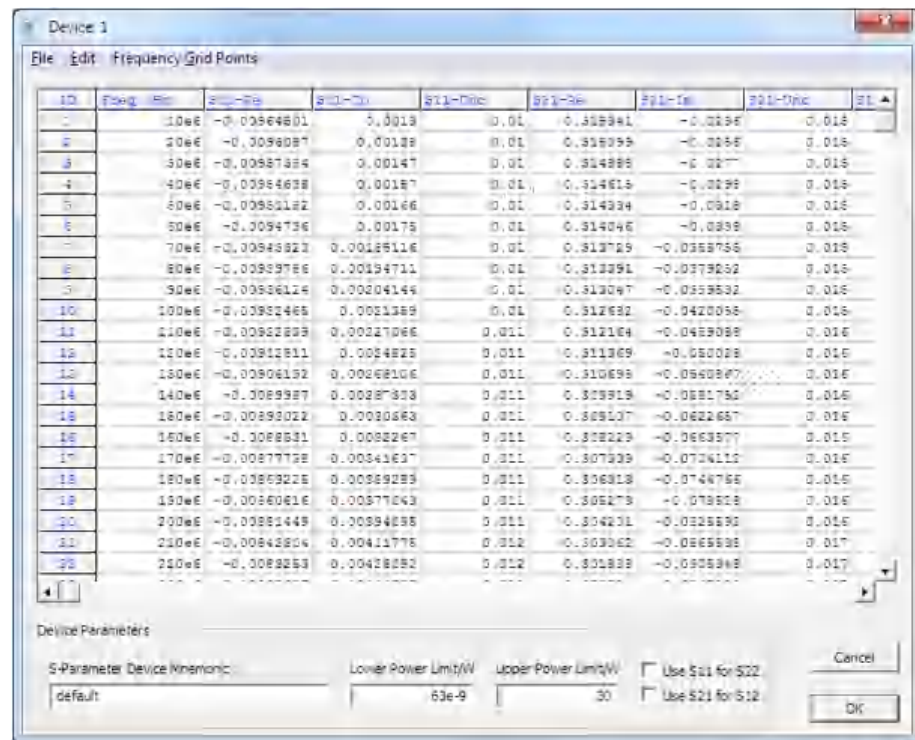


- If needed, load uncertainty data. See ["To load an uncertainty parameter file"](#) on page 91.
- Check the entries in the "S-Parameter Device Mnemonic", "Lower Power Limit/W" and "Upper Power Limit/W" fields
- If necessary, change the entries.
 For example, the lower and upper power limits are deduced from the power limits of the power sensor itself and the minimum attenuation of the S-parameter device. If the "Upper Power Limit/W" entry is higher than the power dissipation rating of the attenuator, reduce it accordingly.
- Click "OK" to apply the changes.
 If you want to discard the change, click "Cancel".

To load an uncertainty parameter file

- In the device table, double-click an entry. See also ["Device table"](#) on page 88.
- Select "File" > "Import uncertainties".
- Select the file you want to import.
- Confirm with "Open".

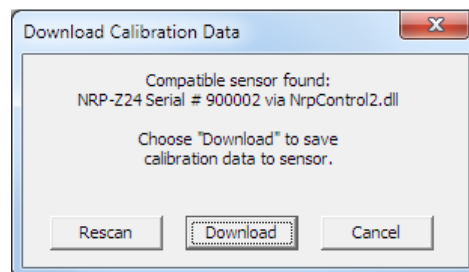
The data from the selected file is loaded in the device table.



To save the calibration data on the power sensor

1. Select "Sensor" > "Save Calibration Data".

The "Download Calibration Data" dialog opens.



2. Confirm that the correct power sensor is selected by clicking "Download".

After a successful transfer of the data to the power sensor, a confirmation message is displayed.

The power sensor can be used with the new S-parameter data.

S2P measurement data files

S2P files are human-readable text files that contain header information and the complex S-parameters of the device under test in columns. This chapter briefly describes the format of the S2P file.

An S2P measurement data file has the following structure (square brackets indicate that the enclosed content is optional):

- **Option line**

The option line has the format $\#[<frequency\ unit>][<parameter>][<format>][<R\ n>]$, where:

- #
Identifies the option line.
- $<frequency\ unit>$
Possible values are Hz, kHz, MHz or GHz. If a frequency unit is not specified, GHz is implicitly assumed.
- $<parameter>$ For S-parameter files. If a parameter is not specified, S is implicitly assumed.
- $<format>$
Possible values are MA (linear magnitude and phase in degree), DB (magnitude in dB, phase in degree) or RI (real and imaginary part). If a format is not specified, MA is implicitly assumed.
- $<R\ n>$
R is optional and followed by the reference impedance in Ω . If no entry is made, R50 is implicitly assumed.

The option line therefore reads:

$\#[HZ | KHZ | MHZ | GHZ] [S] [MA | DB | RI] [R\ 50]$.

- **Measurement frequencies**

The measurement frequencies are listed in ascending order and are specified as follows:

$f_i\ s_{11}(f_i)\ s_{21}(f_i)\ s_{12}(f_i)\ s_{22}(f_i)$

where f_i is the i-th frequency and $s_{jk}(f_i)$ is the display format as specified in the option line:

- $|s_{jk}(f_i)|arg\ s_{jk}(f_i)$
Display format for linear magnitude and phase in degree (MA)
- $20.lg|s_{jk}(f_i)|\ arg\ s_{jk}(f_i)$
Display format for magnitude in dB and phase in degree (DB)
- $Re|s_{jk}(f_i)|Im|s_{jk}(f_i)|$
Display format for real and imaginary part (RI)

- **Comments**

Any line starting with an exclamation mark (!) is interpreted as a comment line.

Uncertainty data files

An uncertainty data file has the following structure (square brackets indicate that the enclosed content is optional):

- **Option line**

The option line has the format $\#[<frequency\ unit>]<parameter>[<format>][<R\ n>]$, where:

- #
Identifies the option line.
- $<frequency\ unit>$
Possible values are Hz, kHz, MHz or GHz. If a frequency unit is not specified, GHz is implicitly assumed.

- *<parameter>*
U must be specified for uncertainty data files. If a parameter is not specified, S is implicitly assumed and as a result an error message is triggered.
- *<format>*
This value is ignored in uncertainty measurement files. The entry is therefore irrelevant.
- *<R n>*
R is optional and followed by the reference impedance in Ω . If no entry is made, R50 is implicitly assumed.

The option line therefore reads:

[HZ | KHZ | MHZ | GHZ] U [MA | DB | RI] [R 50].

- **Measurement frequencies**

The measurement frequencies are listed in ascending order and are specified as follows:

f_i *unc*[$s_{11}(f_i)$] *unc*[$s_{21}(f_i)$] *unc*[$s_{12}(f_i)$] *unc*[$s_{22}(f_i)$]

where f_i is the i-th frequency and *unc*[$s_{jk}(f_i)$] is the uncertainty of the S-parameters that is forwarded as follows:

- As extended absolute uncertainty ($k = 2$) for the magnitude of reflection parameters s_{11} and s_{22}
- As extended uncertainties ($k = 2$) in dB for the magnitude of transmission parameters s_{21} and s_{12}

- **Comments**

Any line starting with an exclamation mark (!) is interpreted as a comment line.

8.8 Configuring measurement results

See also:

- [Chapter 8.9, "Querying measurement results"](#), on page 96

8.8.1 Setting the power unit

The UNIT subsystem contains commands for setting up the power unit.

UNIT:POWer <unit>

Sets the output unit for the measured power values.

Parameters:

<unit> DBM | W | DBUV
*RST: W

Example: UNIT:POW DBM

8.8.2 Setting the result format

The `FORMat` subsystem sets the format of numeric data (measured values) that is exchanged between the remote control computer and the power sensors if high-level measurement commands are used.

Remote commands:

<code>FORMat:BORDER</code>	95
<code>FORMat:SREGister</code>	95
<code>FORMat[:DATA]</code>	95

`FORMat:BORDER` <border>

Selects the order of bytes in 32-bit or 64-bit binary data.

Parameters:

<border> `NORMAL` | `SWAPped`

NORMAL

The 1st byte is the least significant byte, the 4th/8th byte the most significant byte.

Fulfills the Little Endian (little end comes first) convention, used by x86/x64 CPUs, for example.

SWAPped

The 1st byte is the most significant byte, the 4th/8th byte the least significant byte.

Fulfills the Big Endian (big end comes first) convention.

*RST: `NORMAL`

Example: `FORM:BORD NORM`

`FORMat:SREGister` <register>

Specifies which format is used for the return value of `*STB?`.

Parameters:

<register> `ASCIi` | `HEXadecimal` | `OCTal` | `BINary`

*RST: `ASCIi`

Example: `FORM:SREG ASC`

`FORMat[:DATA]` [<data,length>, <length>]

Specifies how the R&S NRPxxP sends the numeric data to the controlling host/computer.

Parameters:

<data,length> <`REAL,32` | `64`>

REAL

Block of binary values, 32-bit or 64-bit each; so-called "SCPI definite length block"

	32 64
	32-bit or 64-bit
	If you omit the length, the R&S NRPxxP sets the last used length.
	Example for REAL, 32 format: #6768000....<binary float values>....
	Example for REAL, 64 format: #71536000....<binary float values>....
<data[,length]>	<ASCIi[,0 to 12]>
	ASCIi
	List of comma-separated, readable values.
	[,0 to 12]
	Defines the number of decimal places.
	The reset value 0 does not restrict the number of decimal places.
	Example for ASCIi, 4 format: 1.2938e-06, -4.7269e-11, ...
	*RST: ASCIi,0

8.9 Querying measurement results

After the measurement, you can query the measurement results.

8.9.1 Continuous average measurement results

FETCh<Sensor>:ARRay[:POWer][:AVG]?.....	96
FETCh<Sensor>[:SCALar][:POWer][:AVG]?.....	96
SENSe<Sensor>[:POWer][:AVG:]BUFFer:DATA?.....	97

FETCh<Sensor>:ARRay[:POWer][:AVG]?

Queries the last valid measurement result of a buffered continuous average measurement.

To configure the measurand, use `CALCulate:FEED` before the measurement is initiated.

Usage: Query only

FETCh<Sensor>[:SCALar][:POWer][:AVG]?

Queries the last valid measurement result of the measurand that was configured before.

To configure the measurand, use `CALCulate:FEED` before the measurement is initiated.

Usage: Query only

SENSe<Sensor>:[POWer:][AVG:]BUFFer:DATA?

Queries the results of the continuous average result buffer and returns them even if the buffer is not full.

In contrast, `FETCh<Sensor>[:SCALar][:POWer][:AVG]?` returns a result only if the buffer is full.

Usage: Query only

8.9.2 Burst average measurement results

FETCh<Sensor>[:SCALar][:POWer]:BURSt?

Queries the last valid measurement result.

To configure the measurand, use `CALCulate:FEED` before the measurement is initiated.

Usage: Query only

8.9.3 Timeslot measurement results

`SENSe<Sensor>:TRACe:DATA?` on page 97

FETCh<Sensor>[:SCALar][:POWer]:TSLot?

Queries the last valid measurement result.

To configure the measurand, use `CALCulate:FEED` before the measurement is initiated.

Usage: Query only

8.9.4 Trace measurement results

`SENSe<Sensor>:TRACe:DATA?`..... 97
`SENSe<Sensor>:TRACe:FFT:DATA?`.....99

SENSe<Sensor>:TRACe:DATA?

Returns the measured trace data in a well-defined format, showing the course of power over a defined time.

Unlike `FETCh<Sensor>[:SCALar][:POWer][:AVG]?`, this command takes the settings of `SENSe<Sensor>:AUXiliary` into account, as explained below.

Command response

Besides the average power, the power sensor can measure additional measurands like minimum, maximum or random. These additional measurands are denoted as auxiliary measurands and are selected by `SENSe<Sensor>:AUXiliary`.

A trace measurement can deliver up to 3 measurands. Therefore, the resulting block of data returned can contain up to 3 blocks of user data.

Basically, the response represents a "definite length arbitrary block response data" as defined in IEEE488.2. This object consists of a header and content. [Figure 8-12](#) outlines the response format:

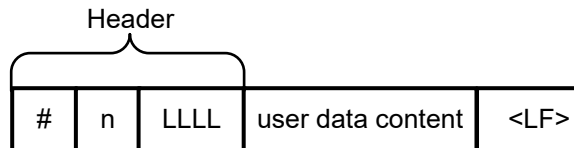


Figure 8-12: Response format

Header	#	Starting character
	n	Single digit that defines how many of the following digits are interpreted as the size of the content.
	LLLLL	Number consisting of as many digits as specified by "n". This number gives the size of the content.
User data content	See also Figure 8-13 . As many bytes as specified by "LLLLL".	
<LF>	Single linefeed character	

Examples

The arbitrary block response data for a user data that contains 45182 bytes is:

```
#545182xxxxxxx.....xxxxxxx <LF>
```

The arbitrary block response data for a user data content 'THIS IS A TEST' is:

```
#214THIS IS A TEST<LF>
```

Explanation: 'THIS IS A TEST' has 14 bytes, and '14' has 2 digits, hence the #214

User data content

In the further description, the term "user data content" is used for the totality of the contained measurement results.

In the user data content, there are similar mechanisms as with arbitrary block response data. As indicated above, the user data content can have one or more blocks with trace measurement results, depending on the selection of auxiliary measurands. Each section is composed of:

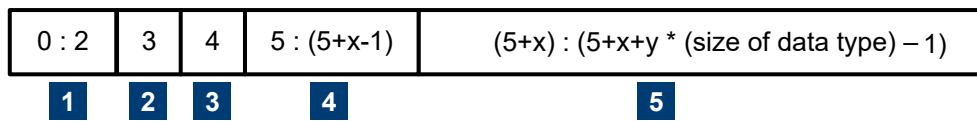


Figure 8-13: User data content format (byte)

1	<p>Result type</p> <p>Always 3 bytes, one for AVG, one for MIN and one for MAX or RND</p>
2	<p>Data type</p> <p>Designator for the contained data type with the size of 1 byte. Currently, the only possible designator is "f" for 4-byte IEEE754 float data type, little endian.</p>
3	<p>Single digit that defines how many of the following digits are interpreted as the number of contained float values.</p>
4	<p>User data length</p> <p>Number consisting of as many digits as specified by (3). This number gives the number of contained float values contained in the user data.</p>
5	<p>User data</p> <p>Measurement result values in the format that is described by the data type. Currently IEEE754 float only.</p>

If no `SENSe<Sensor>:AUXiliary` measurands have been activated before executing the measurement, the user data content is finished here. In case that auxiliary measurands have been selected, the above section is repeated for every auxiliary measurand. The user data content looks like:

```
AVGf3100... (400byte AVG values)...MINf3100... (400byte min.
values)...MAXf3100... (400byte max. values)...
```

Where each of

```
... (400byte AVG values)...
```

```
... (400byte min. values)...
```

```
... (400byte max. values)...
```

Stands for 400 bytes as the equivalent of 100 float values.

The user data content is embedded in the arbitrary block response data response.

Example: `TRAC:DATA?`

Usage: Query only

SENSe<Sensor>:TRACe:FFT:DATA?

Returns a power spectrum of the measured trace data, showing the course of power over a frequency range.

The fast Fourier transform (FFT) of the measured trace data is based on the following parameters:

- Window function
Set by `SENSe<Sensor>:TRACe:FFT:WINDow`.
- Number of values used
The number of required values per trace sequence is set by `SENSe<Sensor>:TRACe:POINts`. For computing the FFT, the closest power of two below this number is used, 2^N . For example, if the set number is 300, then 256 values are used in the FFT, 2^8 .
- Number of values returned
The number of computed values is half the number of used values.
- Distance of the spectral lines in Hz
Equals the reciprocal of the trace length, which is set by `SENSe<Sensor>:TRACe:TIME`.

Note: The spectrum computed as described is the spectrum of the *envelope power*, different from the results displayed by a spectrum analyzer. For questions on details, contact our support team.

Usage: Query only

8.9.5 Pulse analysis measurement results

For information on the pulse analysis and how to configure the measurement, see [Chapter 8.6.5, "Pulse analysis measurement"](#), on page 67.

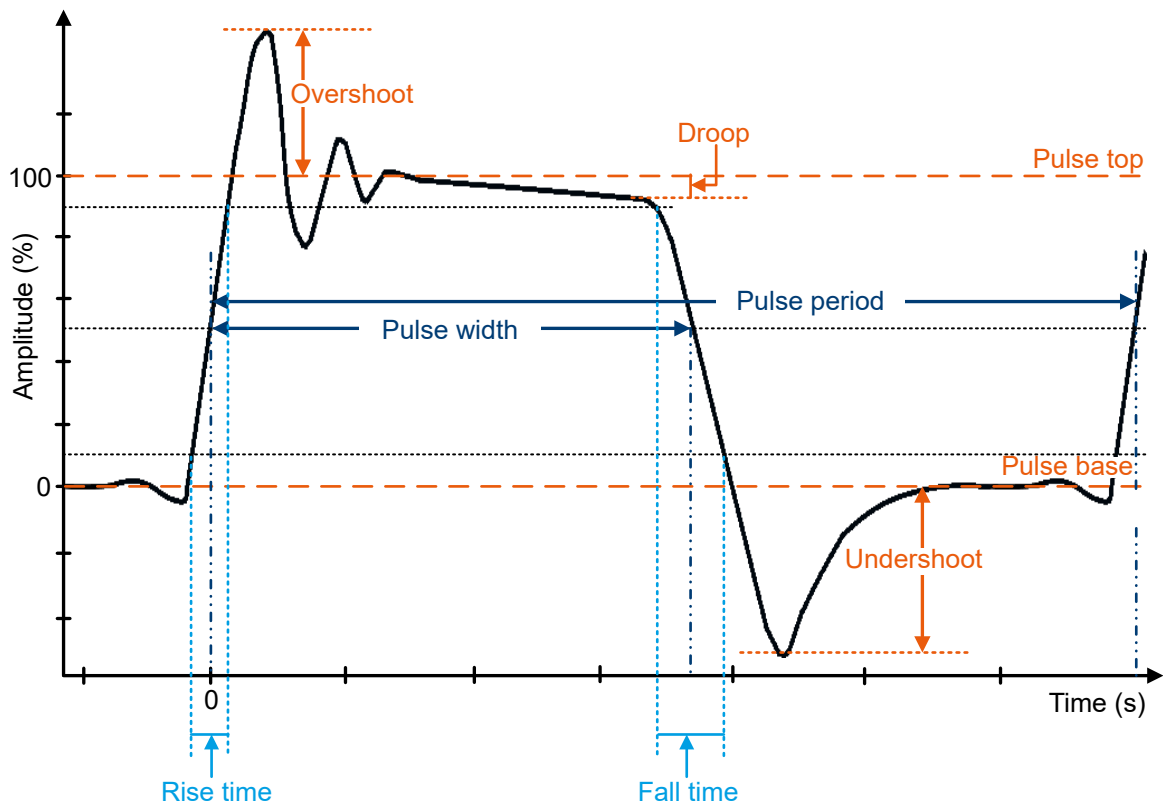


Figure 8-14: Pulse analysis parameters

8.9.5.1 Time-related pulse analysis results

Remote commands:

<code>SENSe<Sensor>:TRACe:MEASurement:OFFSet:TIME</code>	101
<code>SENSe<Sensor>:TRACe:MEASurement:PULSe:DCYClE?</code>	102
<code>SENSe<Sensor>:TRACe:MEASurement:PULSe:DURation?</code>	102
<code>SENSe<Sensor>:TRACe:MEASurement:PULSe:PERiod?</code>	102
<code>SENSe<Sensor>:TRACe:MEASurement:PULSe:SEParation?</code>	103
<code>SENSe<Sensor>:TRACe:MEASurement:TRANSition:NEGative:DURation?</code>	103
<code>SENSe<Sensor>:TRACe:MEASurement:TRANSition:NEGative:OCCurrence?</code>	103
<code>SENSe<Sensor>:TRACe:MEASurement:TRANSition:POSitive:DURation?</code>	103
<code>SENSe<Sensor>:TRACe:MEASurement:TRANSition:POSitive:OCCurrence?</code>	104
<code>SENSe<Sensor>:TRACe:MEASurement:TRANSition:SPERiod?</code>	104

`SENSe<Sensor>:TRACe:MEASurement:OFFSet:TIME` <time>

Sets the start time of the pulse analysis, referenced to delay set by `TRIGger:DELay`.

By default, the parameters of the first detected pulse are determined. Using this command, you can determine parameters of another pulse. Make sure that the pulse analysis does not begin until shortly before this pulse.

Parameters:

<time> Range: Depends on the trigger delay.
 *RST: 0.0
 Default unit: s

SENSe<Sensor>:TRACe:MEASurement:PULSe:DCYClE?

Queries the duty cycle of the measured power. Unit: %

$$\text{Duty cycle} = 100 \% \times \frac{\text{Pulse width}}{\text{Pulse period}}$$



If the pulse period, and thus the duty cycle, cannot be determined, NaN for "not a number" is returned.

Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:PULSe:DURation?

Queries the time between the first positive edge and the subsequent negative edge of the pulse, where the edges occur at crossings of the mid threshold.



If the pulse period, and thus the duty cycle, cannot be determined, NaN for "not a number" is returned.

Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:PULSe:PERiod?

Queries the time between two consecutive edges of the same polarity in seconds. During this time, the pulse signal completes one cycle.



If the trace covers at least three complete periods, the power sensor automatically detects double pulses and determines their period.

If the pulse period, and thus the duty cycle, cannot be determined, NaN for "not a number" is returned.

Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:PULSe:SEParation?

Queries the time between two pulses. The time is measured between the first negative edge and the subsequent positive edge of the pulse in seconds, where the edges occur at crossings of the mid threshold. During this time, the pulse remains at the pulse base level.



If the pulse period, and thus the duty cycle, cannot be determined, NaN for "not a number" is returned.

Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:TRANSition:NEGative:DURation?

Queries the time that the pulse requires to transition from the pulse top level to the pulse base level. The value is read from the first falling edge in the analysis interval.

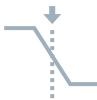


If the pulse period, and thus the duty cycle, cannot be determined, NaN for "not a number" is returned.

Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:TRANSition:NEGative:OCCurrence?

Queries the time when the signal passes through the medial reference power level with falling edge, referenced to the delayed trigger event. Indicates the stop point of the first power pulse within the analysis window.



Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:TRANSition:POSitive:DURation?

Queries the time that the pulse requires to transition from the pulse base level to the pulse top level. The value is read from the first rising edge in the analysis interval.



If the pulse period, and thus the duty cycle, cannot be determined, NaN for "not a number" is returned.

Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:TRANSition:POSitive:OCCurrence?

Queries the time when the signal passes through the medial reference power level with rising edge, referenced to the delayed trigger event. Indicates the start point of the first power pulse within the analysis window.



Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:TRANSition:SPERiod?

Queries the number of samples per second.

Usage: Query only

8.9.5.2 Power-related pulse analysis results

Remote commands:

SENSe<Sensor>:TRACe:MEASurement:POWer:AVG?	104
SENSe<Sensor>:TRACe:MEASurement:POWer:HREFerence?	105
SENSe<Sensor>:TRACe:MEASurement:POWer:LREFerence?	105
SENSe<Sensor>:TRACe:MEASurement:POWer:MAX?	105
SENSe<Sensor>:TRACe:MEASurement:POWer:MIN?	105
SENSe<Sensor>:TRACe:MEASurement:POWer:PULSe:BASE?	105
SENSe<Sensor>:TRACe:MEASurement:POWer:PULSe:DROop?	106
SENSe<Sensor>:TRACe:MEASurement:POWer:PULSe:TOP?	106
SENSe<Sensor>:TRACe:MEASurement:POWer:REFerence?	106
SENSe<Sensor>:TRACe:MEASurement:TRANSition:NEGative:OVERshoot?	106
SENSe<Sensor>:TRACe:MEASurement:TRANSition:POSitive:OVERshoot?	106

SENSe<Sensor>:TRACe:MEASurement:POWer:AVG?

Queries the average power during the time in which the pulse is active.



Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:POWer:HREFerence?

Queries the power level at high reference level set by [SENSe<Sensor>:TRACe:MEASurement:DEFine:TRANSition:HREFerence](#).

Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:POWer:LREFerence?

Queries the power level at low reference level set by [SENSe<Sensor>:TRACe:MEASurement:DEFine:TRANSition:LREFerence](#).

Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:POWer:MAX?

Queries the maximum power measured within the analysis window. Unit: Watts.



Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:POWer:MIN?

Queries the minimum power measured within the analysis window. Unit: Watts.



Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:POWer:PULSe:BASE?

Queries the pulse base power level detected by the selected analysis algorithm, [SENSe<Sensor>:TRACe:MEASurement:ALGorithm](#). This value is used as a reference (0 %) to determine other parameter values such as the rising or falling thresholds.



Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:POWer:PULSe:DROOp?

Queries the droop of the pulse. The droop is the difference between leading edge amplitude, pulse top, and falling edge amplitude, as shown in [Figure 8-14](#). The droop is expressed in percentage of the pulse top amplitude.

The pulse droop measurement is used to test the stability of the output power of a device, e.g. an amplifier or a transmitter.

Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:POWer:PULSe:TOP?

Queries the pulse top power level detected by the selected analysis algorithm, [SENSe<Sensor>:TRACe:MEASurement:ALGorithm](#). This value is used as a reference (0 %) to determine other parameter values such as the rising or falling thresholds.



Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:POWer:REFerence?

Queries the power level at reference level. See [SENSe<Sensor>:TRACe:MEASurement:DEFine:DURation:REFerence](#).

Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:TRANSition:NEGative:OVERshoot?

Queries the height of the local minimum before a rising edge, divided by the pulse amplitude:

The search interval begins when the signal falls below the lower reference level set by [SENSe<Sensor>:TRACe:MEASurement:DEFine:TRANSition:LREFerence](#) and ends half a period thereafter.



Usage: Query only

SENSe<Sensor>:TRACe:MEASurement:TRANSition:POSitive:OVERshoot?

Queries the height of the local maximum before a falling edge, divided by the pulse amplitude:

$$\text{Positive overshoot} = 100 \% \times \frac{\text{Max. power} - \text{pulse top power}}{\text{Pulse amplitude}}$$

The search interval begins when the signal falls below the lower reference level set by `SENSe<Sensor>:TRACe:MEASurement:DEFine:TRANSition:HREFerence` and ends half a period thereafter.



Usage: Query only

8.9.6 Statistics measurement results

You can query the average and peak power. See also [Figure 8-5](#) and [Figure 8-6](#).

<code>SENSe<Sensor>:STATistics:POWer:AVG?</code>	107
<code>SENSe<Sensor>:STATistics:POWer:PEAK?</code>	107
<code>SENSe<Sensor>:STATistics:POWer:PEAK:HOLD</code>	107
<code>SENSe<Sensor>:STATistics:POWer:PEAK:RESet</code>	108

SENSe<Sensor>:STATistics:POWer:AVG?

Queries the average power of the measured signal.

Usage: Query only

SENSe<Sensor>:STATistics:POWer:PEAK?

Queries the peak power value of the current statistics measurement. You can send this query after the measurement or while it is running.

At the start of a new measurement, the peak power value of the previous measurement is deleted. If you want to evaluate the peak power value over several consecutive measurements, use `SENSe<Sensor>:STATistics:POWer:PEAK:HOLD`.

Configure the measurement time using `SENSe<Sensor>:STATistics:TIME`.

Usage: Query only

SENSe<Sensor>:STATistics:POWer:PEAK:HOLD <state>

Enables or disables retaining the peak power value of the previous statistics measurements.

Parameters:

<state>

ON

Retains the peak power value over consecutive measurements.

OFF

At the start of a new measurement, deletes the peak power value of the previous measurement.

*RST: OFF

SENSe<Sensor>:STATistics:POWer:PEAK:RESet

Explicitly deletes the peak power value of the statistics measurement.

Usage: Event

8.10 Calibrating, zeroing

Zeroing removes offset voltages from the analog circuitry of the power sensor, so that there are only low powers displayed if no power applied. The zeroing process can take more than 8 seconds to complete.

Zeroing is recommended if:

- The temperature has varied by more than 5 K.
- The power sensor has been replaced.
- No zeroing was performed in the last 24 hours.
- Signals of very low power are to be measured, for instance, if the expected measured value is less than 10 dB above the lower measurement range limit.



Turn off all test signals before zeroing. An active test signal during zeroing causes an error.

Remote commands:

CALibration:DATA	108
CALibration:DATA:LENGth?	109
CALibration:USER:DATA	109
CALibration:USER:DATA:LENGth?	109
CALibration:ZERO:FAST:AUTO	109
CALibration<Channel>:ZERO:AUTO	110

CALibration:DATA <caldata>

Writes a binary calibration data set in the memory of the power sensor.

Parameters:

<caldata> <block_data>

CALibration:DATA:LENGth?

Queries the length in bytes of the calibration data set currently stored in the flash memory. Programs that read out the calibration data set can use this information to determine the capacity of the buffer memory required.

Example:

```
CAL:DATA:LENG?
Query
57392
Response
```

Usage: Query only

CALibration:USER:DATA <caldata>

Transfers the user calibration data set, which mainly contains S-parameter sets for user-specific devices. The query returns the data as it was downloaded to the power sensor before.

After downloading of a new user calibration data set to the power sensor, the current S-parameter correction settings become invalid. Safe operation of the power sensor is only possible if the `SElect` and `STATe` commands are repeated after download. See also:

- `SENSe<Sensor>:CORRection:SPDevice:STATe`
- `SENSe<Sensor>:CORRection:SPDevice:SElect`

Parameters:

<caldata> <block_data>

CALibration:USER:DATA:LENGth?

Queries the length of the user calibration data block.

Usage: Query only

CALibration:ZERO:FAST:AUTO <mode>

Available in trace and statistics modes.

Performs fast zeroing. Since the commands are processed very quickly, they are not overlapping.

Even though the execution time is shorter than that for standard zeroing by a factor of 100 or more, measurement accuracy is not affected. Fast zeroing is available for the entire frequency range.

Parameters:

<mode> OFF | ONCE
*RST: OFF

CALibration<Channel>:ZERO:AUTO <state>

Performs zero calibration.

Turn off all test signals before zeroing. An active test signal during zeroing causes an error.

While zero calibration is in progress, no queries or other setting commands are allowed, since the command is synchronous. Any communication attempt can run into a timeout.

After zero calibration, query the static error queue (`SYSTem:SERRor?`). The following responses are possible:

- 0
No error, the zero calibration was successful.
- -240
Warning, zero calibration failed. See also the example.

Suffix:

<Channel> 1 to 4
Measurement channel if more than one channel is available.

Parameters:

<state> **ONCE**
Only valid parameter for this command.

0
Return value if no calibration is in progress.

*RST: ONCE

Example:

```
*CLS
CAL1:ZERO:AUTO ONCE
Performs zeroing. Takes several seconds.
SYST:SERR?
Query
-240
Response: Warning; Zero Calibration failed; Results Degrading.
```

8.11 Testing

The selftest allows a test of the internal circuitry of the power sensor.



Do not apply a signal to the power sensor while the selftest is running. If the selftest is carried out with a signal being present, error messages can erroneously be output for the test steps *Offset Voltages* and/or *Noise Voltages*.

TEST:SENSor? [<Item>]

Starts a selftest of the power sensor. In contrast to `*TST?`, this command returns detailed information that you can use for troubleshooting.

Query parameters:	
<Item>	String
Usage:	Query only

8.12 Configuring the system

The `SYSTem` subsystem contains a series of commands for general functions that do not directly affect the measurement.

8.12.1 Configuring general functions

<code>SYSTem:DFPPrint<Channel>?</code>	111
<code>SYSTem:ERRor:ALL?</code>	112
<code>SYSTem:ERRor:CODE:ALL?</code>	112
<code>SYSTem:ERRor:CODE[:NEXT]?</code>	112
<code>SYSTem:ERRor:COUNT?</code>	112
<code>SYSTem:ERRor[:NEXT]?</code>	113
<code>SYSTem:FWUPdate</code>	113
<code>SYSTem:FWUPdate:STATus?</code>	113
<code>SYSTem:HELP:HEADers?</code>	114
<code>SYSTem:INFO?</code>	114
<code>SYSTem:INITialize</code>	114
<code>SYSTem:LANGuage</code>	114
<code>SYSTem:LED:COLor</code>	115
<code>SYSTem:LED:MODE</code>	115
<code>SYSTem:MINPower?</code>	115
<code>SYSTem:PARAmeters?</code>	116
<code>SYSTem:PARAmeters:DELTA?</code>	116
<code>SYSTem:PRESet</code>	116
<code>SYSTem:REBoot</code>	116
<code>SYSTem:REStart</code>	117
<code>SYSTem:RUTime</code>	117
<code>SYSTem:SERRor?</code>	117
<code>SYSTem:SUTime</code>	117
<code>SYSTem:TLEVels?</code>	117
<code>SYSTem:TRANsaction:BEgin</code>	117
<code>SYSTem:TRANsaction:END</code>	118
<code>SYSTem[:SENsor]:NAME</code>	118
<code>SYSTem:VERSion?</code>	118

SYSTem:DFPPrint<Channel>?

Reads the footprint file of the power sensor.

Suffix:	
<Channel>	1...4
	Measurement channel if more than one channel is available.

Usage: Query only

SYSTem:ERRor:ALL?

Queries all unread entries in the SCPI communication error queue and removes them from the queue.

Returns a comma-separated list of error numbers and a short error description in the first-in first-out order.

Example: SYST:ERR:ALL?
Query
0, "No error"
Response

Usage: Query only

SYSTem:ERRor:CODE:ALL?

Queries all unread entries in the SCPI communication error queue and removes them from the queue.

Returns a comma-separated list of error numbers, but no error description.

Example: SYST:ERR:CODE:ALL?
Query
0
Response: No errors have occurred since the error queue was last read out.

Usage: Query only

SYSTem:ERRor:CODE[:NEXT]?

Queries the SCPI communication error queue for the oldest entry and removes it from the queue.

Returns the error number, but no error description.

Example: SYST:ERR:CODE?
Query
0
Response: No errors have occurred since the error queue was last read out.

Usage: Query only

SYSTem:ERRor:COUNT?

Queries the number of entries in the SCPI communication error queue.

Example: `SYST:ERR:COUN?`
 Query
 1
 Response: One error has occurred since the error queue was
 last read out.

Usage: Query only

SYSTem:ERRor[:NEXT]?

Queries the SCPI communication error queue for the oldest entry and removes it from the queue.

Returns an error number and a short description of the error.

Example: `SYST:ERR?`
 Query
 0, 'no error'
 Response: No errors have occurred since the error queue was
 last read out.

Usage: Query only

SYSTem:FWUPdate <fwudata>

Loads new operating firmware into the power sensor. Rohde & Schwarz provides the update file. For further details, see [Chapter 6, "Firmware update"](#), on page 29.

If you want to integrate a firmware update function in an application, see the example given in [Chapter 6.2.2, "Using remote control"](#), on page 31.

Setting parameters:

<fwudata> <block_data>
 Definite length arbitrary block data containing the direct copy of
 the binary *.rsu file in the following format:
 #
 Single digit indicating how many digits follow to specify the size
 of the binary file.
 Number that specifies the size of the binary file.
 Binary data
 0x0a as appended delimiter for line feed

Usage: Setting only

SYSTem:FWUPdate:STATus?

Reads the result of the firmware update performed using [SYSTem:FWUPdate](#) on page 113.

While a firmware update is in progress, the LED of the power sensor flashes in bright white color. When the firmware update is completed, you can read the result.

The result of the query is a readable string.

Example: `SYST:FWUP:STAT?`
 Query
 "Success"
 Response

Usage: Query only

SYSTem:HELP:HEADers? [<Item>]

Returns a list of all SCPI commands supported by the power sensor.

Query parameters:

<Item> <block_data>

Usage: Query only

SYSTem:INFO? [<item>]

Queries information about the power sensor.

If queried without parameters, the command returns all available information in the form of a list of strings separated by commas.

If you want to query specific information, add the query parameter:

`SYST:INFO? "<string>"`

Query parameters:

<item> "Manufacturer", "Type", "Stock Number", "Internal Stock Number", "Serial", "SW Build", "FPGA Build", "Sensor Name", "Technology", "Function", "MinPower", "MaxPower", "MinFreq", "MaxFreq", "Resolution", "Impedance", "Coupling", "Cal. Abs.", "Cal. Refl.", "Cal. S-Para.", "Cal. Misc.", "Cal. Temp.", "Cal. Lin.", "SPD Mnemonic", "TestLimit", "TestLimit pd", "Uptime"

Usage: Query only

SYSTem:INITialize

Sets the power sensor to the standard state, i.e. the default settings for all test parameters are loaded in the same way as with *RST. The power sensor then outputs a complete list of all supported commands and parameters. With the command, the remote-control software can automatically adapt to the features of different types of power sensors with different functionality.

Usage: Event

SYSTem:LANGuage <language>

Selects an emulation of a different command set.

Parameters:

<language> SCPI
 *RST: SCPI

SYSTem:LED:COLor <color>

Effective if the system status LED is user-controlled, [SYSTem:LED:MODE USER](#).

Sets the color and the flash code of the system status LED.

Parameters:

<color> Hexadecimal code described as
 0x0krrggbb
 With
 k = 0: steady on; k = 1: slowly flashing; k = 2: fast flashing
 rr = red
 gg = green
 bb = blue
 Range: 0x00 to 0x02FFFFFF
 *RST: 0x00A0A0A0

Example:

SYST:LED:MODE USER
 The system status LED is user-controlled.

SYST:LED:COL #H01a00000
 The LED flashes slowly in red.

SYSTem:LED:MODE SENSOor
 The system status LED is controlled by the power sensor firm-
 ware.

SYSTem:LED:MODE <mode>

Sets whether the color of the system status LED is controlled by the power sensor firm-
 ware or by the user settings.

For more information, see [SYSTem:LED:COLor](#).

Parameters:

<mode> USER | SENSOor
 *RST: SENSOor

SYSTem:MINPower?

Queries the lower power measurement limit.

This value changes if [SENSe<Sensor>:CORRection:SPDevice:STATe](#) is set to ON. The lower measurement limit refers to the power sensor or to the combination of a power sensor and the components connected ahead of it.

Use this query to determine a useful resolution for the result display near the lower measurement limit.

Usage: Query only

SYSTem:PARAmeters?

Returns an XML-output containing all commands with the following information, if available for the command:

- Default value
- Minimum value
- Maximum value
- Parameters
- Limits

Each command is shortened to a command token, consisting only of the mnemonics short form. For example, `CALibration:DATA` is shortened to `CALDATA` as command token.

Usage: Query only

SYSTem:PARAmeters:DELTA?

Returns an XML-output containing all commands that differ from the defined default status set by `*RST` on page 38.

The commands are accompanied by the same information as for `SYSTem:PARAmeters?`.

Usage: Query only

SYSTem:PRESet

Resets the power sensor.

The command essentially corresponds to the `*RST` command, with the exception that the settings of the following commands are persistently held:

```
INITiate:CONTinuous  
SENSe:AVERAge:TCONtrol
```

Usage: Event

SYSTem:REBoot

Reboots the power sensor.

Usage: Event

SYSTem:REStart

Restarts the firmware of the power sensor.

Usage: Event

SYSTem:RUTime <update_time>

Effective only in the NRP legacy mode. Relevant only in continuous measurement mode, [INITiate:CONTinuous ON](#).

Sets the result update time. That is the maximum rate in which the power sensor can output measurement results.

Parameters:

<update_time> Range: 0.0 to 10.0
 *RST: 0.1
 Default unit: Seconds

SYSTem:SERRor?

Queries the next static error, if available.

Static errors, as a rule, prevent the execution of normal measurements.

Errors in SCPI communication are queried using [SYSTem:ERRor\[:NEXT\]?](#).

Usage: Query only

SYSTem:SUTime <update_time>

Effective only in the NRP legacy mode. Relevant only in continuous measurement mode, [INITiate:CONTinuous ON](#).

Sets the status update time. That is the maximum rate in which the power sensor can output measurement results.

Parameters:

<update_time> Range: 0.0 to 10.0
 *RST: 10e-3
 Default unit: Seconds

SYSTem:TLEVels?

Queries the possible power test levels of the power sensor.

Usage: Query only

SYSTem:TRANsaction:BEgIn

Starts a series of settings.

Usage: Event

SYSTem:TRANSACTION:END

Ends a series of settings.

Usage: Event

SYSTem[:SENSor]:NAME <sensorname>

Sets the name of the power sensor according to your requirements. The specified name is displayed in the web user interface of the network power sensors.

The name that you specify here is independent from the hostname of the power sensor. However, if you do not specify a name, the hostname is used as default.

Parameters:

<sensorname>

Example: SYST:NAME "InputModule-X5"

SYSTem:VERSION?

Queries the SCPI version that the command set of the power sensor complies with.

Example: SYST:VERS?
 Query
 1999.0
 Response: SCPI version from 1999.

Usage: Query only

8.13 Using the status register

Further information:

- [Chapter 10.2, "Status reporting system"](#), on page 127

Contents:

- [General status register commands](#)..... 119
- [Reading the CONDition part](#)..... 119
- [Reading the EVENT part](#)..... 119
- [Controlling the ENABLE part](#)..... 120
- [Controlling the negative transition part](#)..... 120
- [Controlling the positive transition part](#)..... 120

8.13.1 General status register commands

STATus:PRESet.....	119
STATus:QUEue[:NEXT]?.....	119

STATus:PRESet

Resets the edge detectors and `ENABLE` parts of all registers to a defined value.

Usage: Event

STATus:QUEue[:NEXT]?

Queries the most recent error queue entry and deletes it.

Positive error numbers indicate power sensor-specific errors. Negative error numbers are error messages defined by SCPI.

If the error queue is empty, the error number 0, "No error", is returned.

Usage: Query only

8.13.2 Reading the CONDition part

STATus:DEVIce:CONDition?
 STATus:OPERation:CALibrating:CONDition?
 STATus:OPERation:CONDition?
 STATus:OPERation:MEASuring:CONDition?
 STATus:OPERation:SENSe:CONDition?
 STATus:OPERation:TRIGger:CONDition?
 STATus:QUESTionable:CALibration:CONDition?
 STATus:QUESTionable:CONDition?
 STATus:QUESTionable:POWer:CONDition?
Usage: Query only

8.13.3 Reading the EVENT part

STATus:DEVIce[:EVENT]?
 STATus:OPERation:CALibrating[:SUMMARY][:EVENT]?
 STATus:OPERation[:EVENT]?
 STATus:OPERation:MEASuring[:SUMMARY][:EVENT]?
 STATus:OPERation:SENSe[:SUMMARY][:EVENT]?
 STATus:OPERation:TRIGger[:SUMMARY][:EVENT]?
 STATus:QUESTionable:CALibration[:SUMMARY][:EVENT]?
 STATus:QUESTionable[:EVENT]?
 STATus:QUESTionable:POWer[:SUMMARY][:EVENT]?
Usage: Query only

8.13.4 Controlling the ENABLE part

STATus:DEvIce:ENABle <value>
STATus:OPERation:CALibrating:ENABle <value>
STATus:OPERation:ENABle <value>
STATus:OPERation:MEASuring:ENABle <value>
STATus:OPERation:SENSe:ENABle <value>
STATus:OPERation:TRIGger:ENABle <value>
STATus:QUESTionable:CALibration:ENABle <value>
STATus:QUESTionable:ENABle <value>
STATus:QUESTionable:POWER:ENABle <value>
Parameters:
 <value> *RST: 0

8.13.5 Controlling the negative transition part

STATus:DEvIce:NTRansition <value>
STATus:OPERation:CALibrating:NTRansition <value>
STATus:OPERation:NTRansition <value>
STATus:OPERation:MEASuring:NTRansition <value>
STATus:OPERation:SENSe:NTRansition <value>
STATus:OPERation:TRIGger:NTRansition <value>
STATus:QUESTionable:CALibration:NTRansition <value>
STATus:QUESTionable:NTRansition <value>
STATus:QUESTionable:POWER:NTRansition <value>
Parameters:
 <value> *RST: 0

8.13.6 Controlling the positive transition part

STATus:DEvIce:PTRansition <value>
STATus:OPERation:CALibrating:PTRansition <value>
STATus:OPERation:PTRansition <value>
STATus:OPERation:MEASuring:PTRansition <value>
STATus:OPERation:SENSe:PTRansition <value>
STATus:OPERation:TRIGger:PTRansition <value>
STATus:QUESTionable:CALibration:PTRansition <value>
STATus:QUESTionable:PTRansition <value>
STATus:QUESTionable:POWER:PTRansition <value>
Parameters:
 <value> *RST: 65535

9 Performing measurement tasks - programming examples

If you install the optional software development kit (SDK) of the R&S NRP Toolkit, programming examples are provided. See [Chapter 5.1, "R&S NRP Toolkit"](#), on page 22.

Under Windows, these examples are installed under:

```
C:\ProgramData\Rohde-Schwarz\NRP-Toolkit-SDK\examples
```

This chapter gives programming examples for measurement tasks performed with the R&S NRP power sensors.

9.1 Performing the simplest measurement

The simplest way to obtain a result is to use the following sequence of commands:

```
*RST
INITiate
FETCh?
```

The `*RST` sets the continuous average mode.

`INITiate` initiates the measurement.

After `*RST`, the trigger system is set to `TRIGger:SOURce IMMEDIATE`. That means the power sensor starts measuring when the measurement is started without waiting for a trigger condition.

After the measurement has been completed, `FETCh<Sensor>[:SCALar][:POWER] [:AVG] ?` delivers the result to the output queue from which it can be fetched.

9.2 Performing a buffered continuous average measurement

This example, written in pseudo code, shows how to set up and execute a buffered continuous average measurement.

```
//Select whether using
//   'BUS Trigger'    --> true
// or 'EXT Trigger'  --> false
bool   bUseBUSTrigger = true;

// Use the first NRP series sensor which is found
if ( VI_SUCCESS == SENSOR.openFirstNrpSensor( "USB?::0X0AAD::?*::INSTR" ) )
{
    //Start with a clean state
    SENSOR.write( "*RST" );
```

Performing a buffered continuous average measurement

```
// Auto Averaging OFF and set Average Count = 4
SENSOR.write( "SENS:AVER:COUN:AUTO OFF" );
SENSOR.write( "SENS:AVER:COUN 4" );

// Select the trigger source
if ( bUseBUSTrigger )
{
    // We want to use '*TRG' to trigger a single physical measurement
    SENSOR.write( "TRIG:SOUR BUS" );
}
else
{
    // We get trigger pulses on the external input (SMB-type connector)
    SENSOR.write( "TRIG:SOUR EXT2" );
}
// Auto-Trigger OFF
SENSOR.write( "TRIG:ATR:STAT OFF" );

// Configure a buffered measurement
// Buffer size is randomly selected to 17
SENSOR.write( "SENS:BUFF:SIZE 17" );
SENSOR.write( "SENS:BUFF:STAT ON" );
SENSOR.write( "TRIG:COUN 17" );

// Read out all errors / Clear error queue
SENSOR.query( "SYST:ERR:ALL?", szBuf, sizeof( szBuf ) );
printf( szBuf );

// Start a 'single' buffered measurement
// Since 17 trigger-counts have been configured,
// the 'single' buffered measurement, which becomes
// initiated by INIT:IMM, is not over until
// 17 physical measurements have been triggered
SENSOR.write( "INIT:IMM" );

// The end of a physical measurement can be recognized
// by a transition to 'NOT MEASURING' which is a
// negative transition on bit 1
SENSOR.write( "STAT:OPER:MEAS:NTR 2" );
SENSOR.write( "STAT:OPER:MEAS:PTR 0" );

// Collect 17 physical measurements
for ( int i = 0; i < 17; i++ )
{
    // As a pre-condition: clear the event register by reading it
    int iDummy;
    SENSOR.query( "STAT:OPER:MEAS:EVEN?", &iDummy );
}
```

```
// Trigger a single physical measurement; either by '*TRG'
// command or by an externally supplied pulse on the SMB-type connector
if ( bUseBUSTrigger )
    SENSOR.write( "*TRG" );

// Wait until the measurement is done
int iMeasEvent = 0;
while ( iMeasEvent != 2 )
{
    SENSOR.query( "STAT:OPER:MEAS:EVEN?", &iMeasEvent );
    iMeasEvent &= 2;
}
printf( "Triggered!\n" );
}
// All 17 physical measurement have been executed.
// That means, buffer is full and can be read
SENSOR.query( "FETCH?", szBuf, sizeof( szBuf ) );
printf( szBuf );
}
```

9.3 Performing trace measurements

```
*RST

//Set the sensor's operation mode to trace
SENSE:FUNCTion "XTIME:POWer"

//Set the carrier frequency
SENSE:FREQuency 1.8e9

//Set the number of points for the trace measurement
//Using 500 points usually represents a good compromise
//between USB transfer speed and resolution
SENSE:TRACe:POINTs 500

//Set the trace time.It influences the time length of a point since each point
//represents the time period resulting from the trace time divided by the
//number of points
SENSE:TRACe:TIME 20e-3

//Set the trace offset time to delay the start point
//of the trace measurement for the specified time
SENSE:TRACe:OFFSet:TIME 50e-6

//Configure the trigger
TRIGger:SOURce INTernal
```

Trace measurement with synchronization to measurement complete

```

TRIGger:SLOPe POSitive
TRIGger:DTIME 0.001
TRIGger:HYSTeresis 0.1
TRIGger:LEVel 30e-6

//Enable and configure the averaging filter
SENSe:TRACe:AVERAge:COUNT 8
SENSe:TRACe:AVERAge:STATe ON

//Select the data output format
FORMat:DATA REAL

//Initiate the measurement
INITiate

//Query the measurement results
FETCh?

```

9.4 Trace measurement with synchronization to measurement complete

This example, written in pseudo code, shows how to set up and execute a trace measurement using a non-blocking technique.

The advantage of using the `FETCH?` command (as shown in the previous example) is, that `FETCH?` waits (blocks) until a measurement result is available. However, this behavior can lead to situations where an application blocks for a longer time (until timeout). For example, if a trigger is missing and thus no results are ever becoming available.

For certain applications, especially interactive ones, it is not the desired behavior that you have to wait until a (probably long) timeout occurs. In these cases, start a measurement and then enter a loop to poll the power sensor until the measurement is ready and the results can safely be retrieved. For such applications, it is recommended to use the status system of the power sensor to find out whether the measurement is ready. The advantage of this approach is that the polling loop can be exited/canceled at any time and the application stays operable (i. e. does not block).

```

// basic setup, similar to the previous example
write( "*RST" );
write( "SENS:FUNC \"XTIM:POW\"" );
write( "SENS:FREQ 1.8e9" );
write( "SENS:TRAC:POIN 500" );
write( "SENS:TRAC:TIME 20e-3" );
write( "TRIG:SOUR INT" );
write( "TRIG:SLOP POS" );
write( "TRIG:DTIM 0.001" );
write( "TRIG:HYST 0.1" );
write( "TRIG:LEV 30e-6" );

```

Trace measurement with synchronization to measurement complete

```
write( "SENS:TRAC:AVER:COUN 8" );
write( "SENS:TRAC:AVER:STAT ON" );

// configuring the event system to recognize the
// end of measurement (i.e. a negative transition
// of bit 1 in the meas operation register)
write( "STAT:OPER:MEAS:NTR 2" );
write( "STAT:OPER:MEAS:PTR 0" );

// resetting the event information by an initial readout
int iEvent = 0;
query( "STAT:OPER:MEAS:EVEN?", &iEvent );

// Now starting the measurement
write( "INIT:IMM" );

bool bMeasReady = false;

// poll until measurement is ready...
// (this loop could also check for cancel-requests
// from the user or other break conditions)
while ( ! bMeasReady )
{
    query( "STAT:OPER:MEAS:EVEN?", &iEvent );
    bMeasReady = ((iEvent & 0x02) != 0);

    if ( ! bMeasReady )
        sleep( 1 );
}

if ( bMeasReady )
{
    query( "SENS:TRAC:DATA?", bufResult );

    // further process the result in 'bufResult'...
    // ::
    // ::
}
```

10 Remote control basics

For general information on remote control of Rohde & Schwarz products via SCPI, refer to www.rohde-schwarz.com/rc-via-scp.

10.1 Remote control interfaces and protocols

For remote control, communication between the R&S NRPxxP power sensors and the controlling host is established based on various interfaces and protocols.

Depending on the power sensor type, the power sensors support different interfaces for remote control.

- R&S NRPxxP USB power sensors are always accessed using USB.

Table 10-1 describes the protocols that are supported for each interface.

Table 10-1: Remote control interfaces and protocols

Interface	Supported by	Protocols, VISA ^{*)} address string and Library
USB	All power sensors	<ul style="list-style-type: none"> • USBTMC USB::<vendor id="">:::product ID>:::serial number>:::INSTR]</vendor> • NRP legacy protocol
<p><small>*) VISA is a standardized software interface library providing input and output functions to communicate with instruments. A VISA installation on the controller is a prerequisite for remote control over USBTMC and LAN (when using VXI-11 or HiSLIP protocol) interfaces.</small></p>		

10.1.1 USB interface

For remote control using USB connection, the computer and the power sensors must be connected via the USB interface. A USB connection requires the VISA library to be installed. VISA detects and configures the R&S power sensors automatically when the USB connection is established. Apart from the USBTMC driver (which comes with the installation of the R&S NRP Toolkit), you do not have to install a separate driver.

USB Test & Measurement Class Specification (USBTMC) is a protocol that is built on top of USB for communication with USB devices. It defines class code information of the power sensor that identifies its functionality to load the respective device driver. Using VISA library, it supports service request, triggers and other operations that are commonly found in GPIB devices.

Besides USBTMC, the NRP legacy protocol is available to ensure the compatibility of the R&S NRPxxP power sensors with the R&S NRP-Z series of power sensors. The usage of this protocol is not recommended for new applications.

The resource string represents an addressing scheme that is used to establish a communication session with the power sensor. It is based on the power sensor address and some power sensor- and vendor-specific information.

USB Resource String

The syntax of the used USB resource string is:

```
USB::::<product ID>::[::INSTR]
```

where:

- <vendor ID> is the vendor ID for Rohde & Schwarz (0x0AAD)
- <product ID> is the product ID for the Rohde & Schwarz power sensor
- <serial number> is the individual serial number on the name plate at the rear side of the power sensor

Table 10-2: R&S NRPxxP USB product IDs

R&S NRPxxP power sensor	USB product ID
R&S NRP18P	0x0143
R&S NRP40P	0x017C
R&S NRP50P	0x017E

Example:

```
USB::0x0AAD::0x0143::100001
```

0x0AAD is the vendor ID for Rohde & Schwarz.

0x0143 is the product ID for the R&S NRP18P power sensor.

100001 is the serial number of the particular power sensor.

10.2 Status reporting system

The status reporting system stores all information on the current operating state of the sensor, and on errors which have occurred. This information is stored in the status registers and in the error queue. You can query both with the commands of the `STATus` subsystem.

10.2.1 Overview

[Fig.10-1](#) shows the hierarchical structure of information in the status registers.

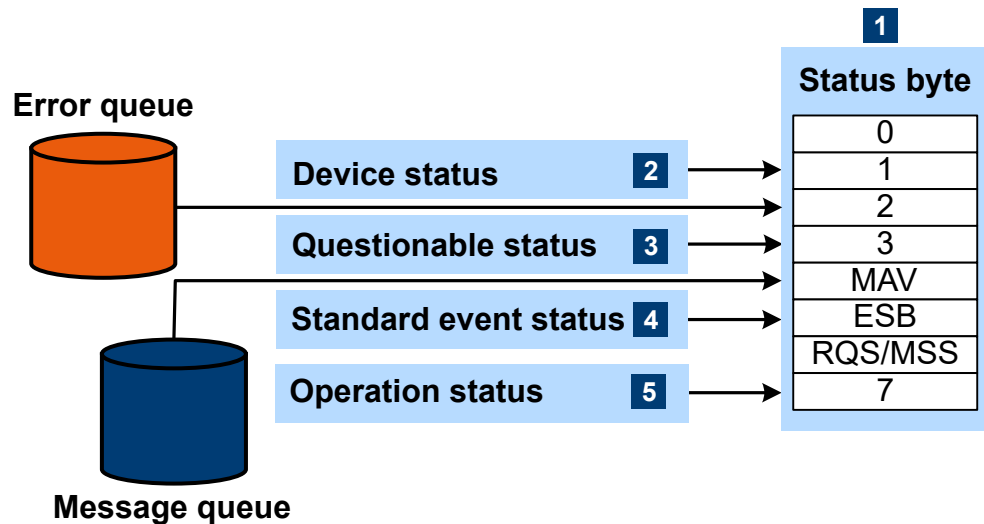


Figure 10-1: Status registers overview

1 = Status byte, see [Table 10-3](#)

2 = [Chapter 10.2.2, "Device status register"](#), on page 129

3 = [Chapter 10.2.3, "Questionable status register"](#), on page 131

4 = [Chapter 10.2.4, "Standard event status and enable register \(ESR, ESE\)"](#), on page 133

5 = [Chapter 10.2.5, "Operation status register"](#), on page 134

The highest level is formed by the status byte register (STB) and the associated service request enable (SRE) register.

The status byte register (STB) receives its information from:

- Standard event status register (ESR)
- Associated standard event status enable register (ESE)
- SCPI-defined operation status register
- Questionable status register, which contains detailed information on the device.

Table 10-3: Used status byte bits and their meaning

Bit no.	Short description	Bit is set if
1	Device status register summary	A sensor is connected or disconnected or when an error has occurred in a sensor, depending on the configuration of the sensor status register. Chapter 10.2.2, "Device status register" , on page 129.
2	Error queue not empty	The error queue has an entry. If this bit is enabled by the service request enable register, each entry of the error queue generates a service request. An error can thus be recognized and specified in detail by querying the error queue. The query yields a conclusive error message. This procedure is recommended since it considerably reduces the problems of IEC/IEEE-bus control.

Bit no.	Short description	Bit is set if
3	Questionable status register summary	An <code>EVENT</code> bit is set in the <code>QUESTIONABLE</code> status register and the associated <code>ENABLE</code> bit is set to 1. A set bit denotes a questionable device status which can be specified in greater detail by querying the questionable status register. Chapter 10.2.3, "Questionable status register" , on page 131.
4	MAV Message available	A readable message is in the message queue. This bit can be used to automate reading of data from the sensor into the controller.
5	ESB Standard event status register summary	One of the bits in the standard event status register is set and enabled in the event status enable register. Setting this bit denotes a serious error which can be specified in greater detail by querying the standard event status register. Chapter 10.2.4, "Standard event status and enable register (ESR, ESE)" , on page 133.
6	RQS/MSS Master status summary	The sensor triggers a service request, which happens if one of the other bits of this register is set together with its enable bit in the service request enable register (SRE).
7	Operation status register summary	An <code>EVENT</code> bit is set in the operation status register and the associated <code>ENABLE</code> bit is set to 1. A set bit denotes that an action is being performed by the sensor. Information on the type of action can be obtained by querying the operation status register. Chapter 10.2.5, "Operation status register" , on page 134.

Further information:

- See [Figure 10-1](#).
- Set and read the service request enable register using `*SRE`.

10.2.2 Device status register

Shows whether static errors (SERR) exist and other power sensor status information.

Sum of bits 1 to 4	0
Measurement not possible	1
Erroneous result	2
Warning	3
Critical	4
0	5
0	6
Legacy locked state	7
0	8
0	9
0	10
0	11
0	12
0	13
0	14
0	15

Figure 10-2: Device status register

Querying the register:

- `STATus:DEvice:CONDition?`
- `STATus:DEvice[:EVENT]?`

Querying the static errors:

- `SYSTem:SERRor?`

Table 10-4: Used device status bits and their meaning

Bit no.	Short description	Bit is set if
0	Sum of bits 1 to 4	Sum/combination of static error (SERR) bits 1 to 4.
1	Measurement not possible	
2	Erroneous results	Static error exists. The measurement result is possibly incorrect.
3	Warning	Static error exists. Status LED of the power sensor is slowly flashing red.

Bit no.	Short description	Bit is set if
4	Critical	Critical static error exists. Status LED of the power sensor is fast flashing red.
7	Legacy locked state	<p>The power sensor is locked in the NRP legacy mode. Via the SCPI channels (USBTMC or TCP/IP), only query commands can be sent, but no setting commands.</p> <p>When the first setting command is sent, the NRP legacy interface takes precedence over all other command channels. This bit is set to 1, and all other channels can only execute query commands. If a setting command is sent via a different channel, the power sensor indicates an error:</p> <p><i>-200,"Execution error; sensor in LEGACY mode"</i></p> <p>To leave the NRP legacy mode, close the NRP legacy channel. Either close the application which opened the NRP legacy channel or close at least the connection to the power sensor.</p>

10.2.3 Questionable status register

Contains information on questionable sensor states that occur if the power sensor is not operated in compliance with its specifications.

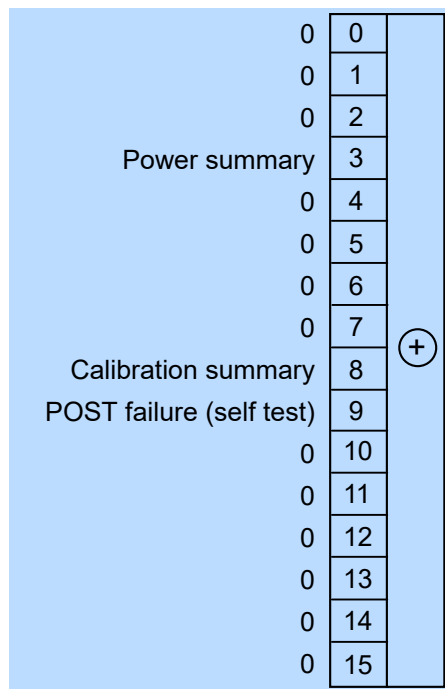


Figure 10-3: Questionable status register

Querying the register:

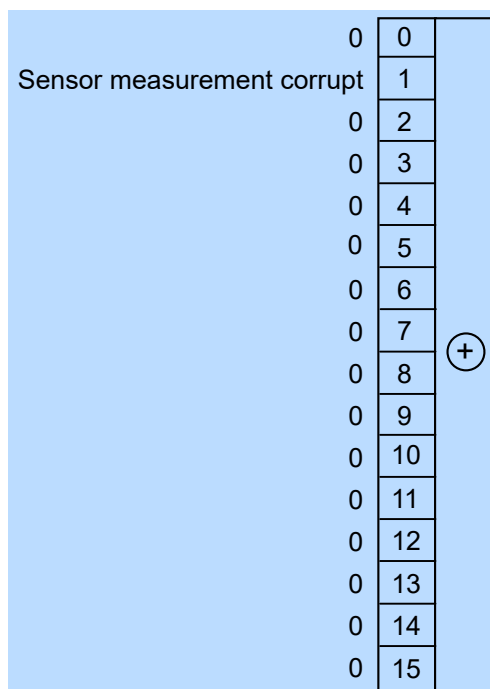
- `STATUS:QUESTIONABLE:CONDITION?`
- `STATUS:QUESTIONABLE[:EVENT]?`

Table 10-5: Used questionable status bits and their meaning

Bit no.	Short description	Bit is set if
3	Power summary	Summary of Questionable power status register exists.
8	Calibration summary	Summary of Questionable calibration status register exists.
9	POST failure (self test)	Built-in test of the R&S NRPxxP that is carried out automatically upon power-up has generated an error.

10.2.3.1 Questionable power status register

Contains information whether the measured power values are questionable.

**Figure 10-4: Questionable power status register**

Querying the register:

- `STATUS:QUESTIONABLE:POWER:CONDITION?`
- `STATUS:QUESTIONABLE:POWER[:SUMMARY][:EVENT]?`

Table 10-6: Used questionable power status bits and their meaning

Bit no.	Short description	Bit is set if
1	Sensor measurement corrupt	Measurement data of the power sensor is corrupt.

10.2.3.2 Questionable calibration status register

Contains information whether the zeroing of the power sensor was successful.

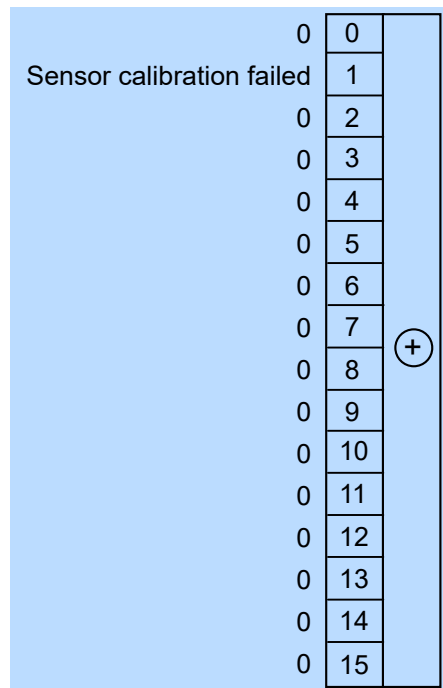


Figure 10-5: Questionable calibration status register

Querying the register:

- `STATUS:QUESTIONABLE:CALIBRATION:CONDITION?`
- `STATUS:QUESTIONABLE:CALIBRATION[:SUMMARY] [:EVENT]?`

Table 10-7: Used questionable calibration status bits and their meaning

Bit no.	Short description	Bit is set if
1	Sensor calibration failed	Zeroing of the power sensor was not successful.

10.2.4 Standard event status and enable register (ESR, ESE)

The `ESR` is already defined in the IEEE 488.2 standard. It is comparable to the `EVENT` register of a SCPI register. The standard event status register can be read out by `*ESR?`.

The `ESE` forms the associated `ENABLE` register. It can be set and read by `*ESE`.

Operation Complete	0	+
0	1	
Query Error	2	
Device-Dependent Error	3	
Execution Error	4	
Command Error	5	
User Request	6	
Power On	7	

Figure 10-6: Standard event status register (ESR)

Table 10-8: Used standard event status bits and their meaning

Bit no.	Short description	Bit is set if
0	Operation complete	All previous commands have been executed and *OPC is received.
2	Query error	The controller wants to read data from the sensor but has not sent a query, or it sends new commands to the sensor before it retrieves existing requested data. A frequent cause is a faulty query which cannot be executed.
3	Device-dependent error	A sensor-dependent error occurs. An error message with a number between -300 and -399 or a positive error number denoting the error in greater detail is entered in the error queue.
4	Execution error	The syntax of a received command is correct but the command cannot be executed due to various marginal conditions. An error message with a number between -200 and -300 denoting the error in greater detail is entered in the error queue.
5	Command error	An undefined command or a command with incorrect syntax is received. An error message with a number between -100 and -200 denoting the error in greater detail is entered in the error queue.
6	User request	The sensor is switched over to manual control.
7	Power on	The sensor is switched on.

10.2.5 Operation status register

Contains information on current operations, CONDition register, or operations performed since the last query, EVENT register.

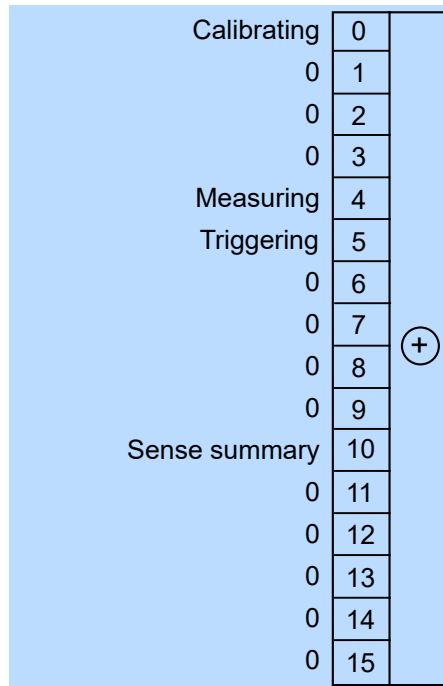


Figure 10-7: Operation status register

Querying the register:

- `STATUS:OPERation:CONDition?`
- `STATUS:OPERation[:EVENT]?`

Table 10-9: Used operation status bits and their meaning

Bit no.	Short description	Bit is set if
0	Calibrating	Summary of Operation calibrating status register exists.
4	Measuring	Summary of Operation measuring status register exists.
5	Triggering	Summary of Operation trigger status register exists.
10	Sense summary	Summary of Operation sense status register exists.

10.2.5.1 Operation calibrating status register

The `CONDition` register contains information whether a power sensor is being calibrated. The `EVENT` register contains information whether a calibration was started or completed since the last query.

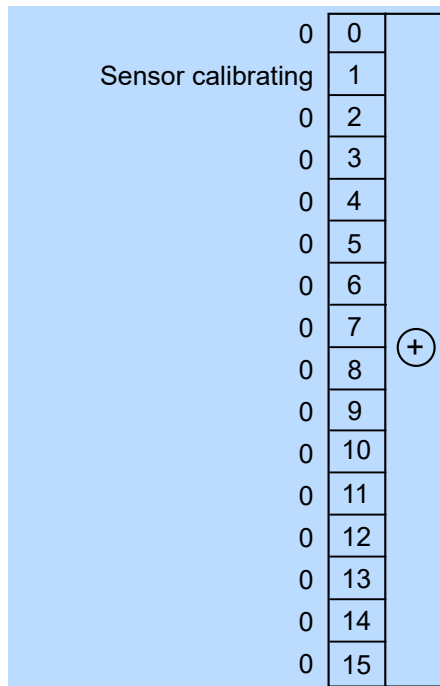


Figure 10-8: Operation calibrating status register

Querying the register:

- `STATUS:OPERation:CALibrating:CONDition?`
- `STATUS:OPERation:CALibrating[:SUMMARY][:EVENT]?`

Table 10-10: Used operation calibrating status bits and their meaning

Bit no.	Short description	Bit is set if
1	Sensor calibrating	Power sensor is being calibrated.

10.2.5.2 Operation measuring status register

The `CONDition` register contains information whether a power sensor is measuring. The `EVENT` register contains information whether a measurement was started or completed since the last query.

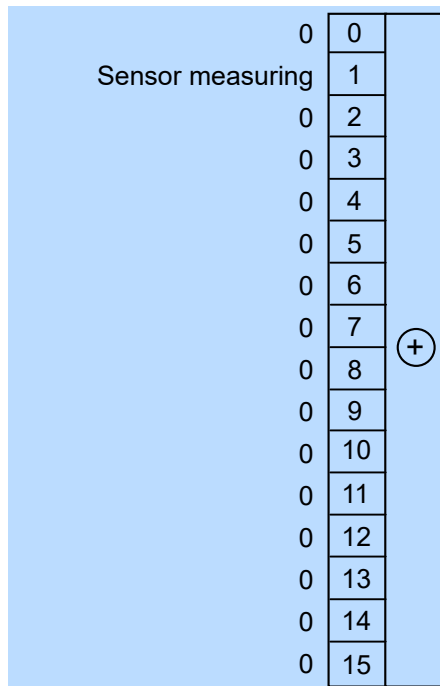


Figure 10-9: Operation measuring status register

Querying the register:

- `STATUS:OPERation:MEASuring:CONDition?`
- `STATUS:OPERation:MEASuring[:SUMMARY][:EVENT]?`

Table 10-11: Used operation measuring status bits and their meaning

Bit no.	Short description	Bit is set if
1	Sensor measuring	Power sensor is measuring.

10.2.5.3 Operation trigger status register

The `CONDition` register contains information whether a power sensor is waiting for a trigger event. The `EVENT` register contains information whether the power sensor has been waiting for a trigger event since the last query.

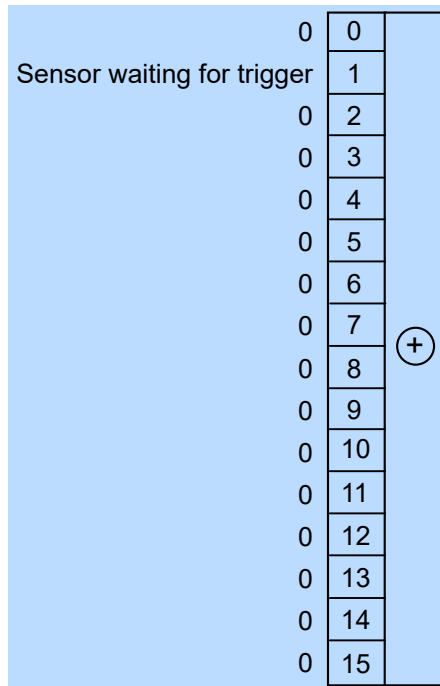


Figure 10-10: Operation trigger status register

Querying the register:

- `STATus:OPERation:TRIGger:CONDition?`
- `STATus:OPERation:TRIGger[:SUMMary][:EVENT]?`

Table 10-12: Used operation trigger status bits and their meaning

Bit no.	Short description	Bit is set if
1	Sensor waiting for trigger	Power sensor is waiting for a trigger event. When the trigger event occurs, the power sensor changes into the measuring state.

10.2.5.4 Operation sense status register

The `CONDition` register contains information whether a power sensor is being initialized. The `EVENT` register contains information whether an initialization was started or completed since the last query.

A power sensor is initialized if:

- Supply voltage is switched on (power-up).
- Power sensor was connected.
- Reset was performed using:
 - `*RST`
 - `SYSTem:PRESet`

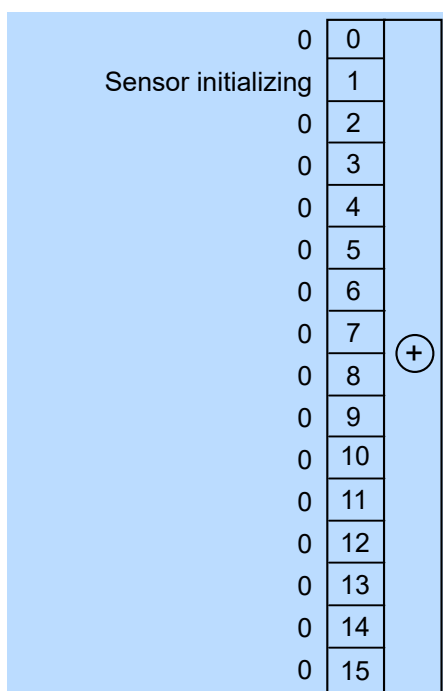


Figure 10-11: Operation sense status register

Querying the register:

- `STATUS:OPERation:SENSe:CONDition?`
- `STATUS:OPERation:SENSe[:SUMMARY][:EVENT]?`

Table 10-13: Used operation sense status bits and their meaning

Bit no.	Short description	Bit is set if
1	Sensor initializing	Power sensor is being initialized.

11 Troubleshooting

- [Displaying status information](#)..... 140
- [Performing a selftest](#)..... 140
- [Problems during a firmware update](#)..... 140
- [Contacting customer support](#)..... 141

11.1 Displaying status information

Status information is available in several ways.

Status LED of the R&S NRPxxP

The position of the status LED is indicated in [Chapter 4, "Power sensor tour"](#), on page 20.

The meaning of the different colors and blinking frequencies is explained in [Chapter 4.2, "Status information"](#), on page 21.

11.2 Performing a selftest

The selftest gives you detailed information that you can use for troubleshooting.



Do not apply a signal to the power sensor while the selftest is running. If the selftest is carried out with a signal being present, error messages can erroneously be output for the test steps *Offset Voltages* and/or *Noise Voltages*.

Using remote control

- ▶ For a quick check, send `TEST:SENSor?`.
- For each test step, PASS or FAIL is listed.

11.3 Problems during a firmware update

The firmware update is described in [Chapter 6, "Firmware update"](#), on page 29.

Solutions for potential problems that can occur when using the Firmware Update for NRP Family, see ["Troubleshooting"](#) on page 31.

Firmware update was interrupted

If for example, a power cut happened during the firmware update, problems can occur.

1. Perform the firmware update again. Sometimes, a further update fixes the problems.

2. If the power sensor is not accessible anymore, contact the service.

Firmware update was aborted

If there is not enough free storage space, the firmware update is aborted. An error message is displayed, and the status LED of the power sensor starts flashing red.

1. Perform a sanitization procedure, as described in the instrument security procedures. This document is available on the product page, see [Chapter 2.1, "Documentation overview"](#), on page 10, or the CD-ROM delivered with the power sensor.
2. Perform the firmware update again.

11.4 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 11-1: QR code to the Rohde & Schwarz support page

12 Transporting

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. You can also contact your local Rohde & Schwarz service center for advice.

Securing

When moving the product in a vehicle or using transporting equipment, make sure that the product is properly secured. Only use items intended for securing objects.

Transport altitude

Unless otherwise specified in the data sheet, the maximum transport altitude without pressure compensation is 4500 m above sea level.

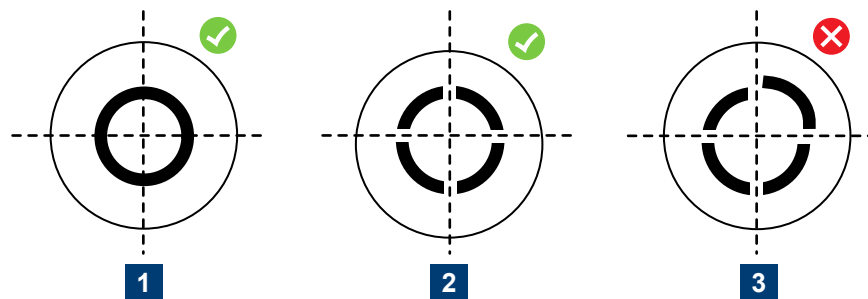
13 Maintenance, storage and disposal

It is advisable to check the nominal data from time to time.

13.1 Regular checks

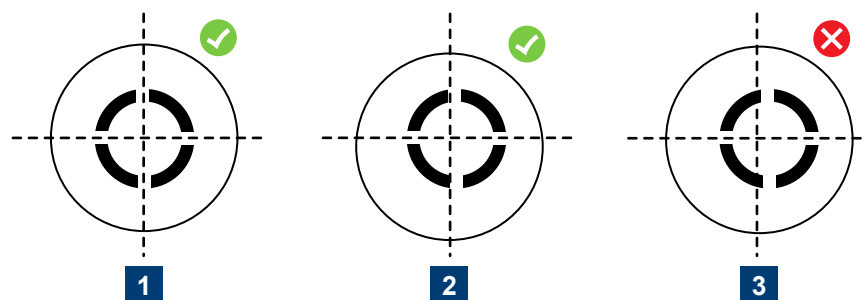
If the power sensor is used frequently, check the RF connectors for visible damage - bent inner conductors, broken contact springs and so on. If the RF connectors are damaged, stop using the power sensor. Contact Rohde & Schwarz customer service, see [Chapter 11.4, "Contacting customer support"](#), on page 141.

Checking the integrity of the RF connector



- 1 = Perfectly formed contacts
- 2 = Normal wear
- 3 = Damaged contacts

Checking the concentricity of the RF connector



- 1 = Perfectly concentric
- 2 = Slightly off-center
- 3 = Excessively off-center

13.2 Cleaning

1. Disconnect the R&S NRPxxP:
 - a) From the DUT.
 - b) From the computer or base unit.
2. Clean the outside of the R&S NRPxxP using a lint-free cloth. You can dampen the cloth with water but keep in mind that the casing is not waterproof. If you use isopropyl alcohol or ethanol instead of water, be careful not to damage the labeling. Do not use cleaning agents that can damage the instrument such as solvents, acids or alkalis.
3. Clean the RF connectors as follows:
 - a) Dislodge any particles using solvent-free compressed air.
 - b) Clean the inside and the connector threads using a foam swab or lint free cloth dampened with isopropyl alcohol or ethanol.

Further information:

- ["To disconnect from the DUT"](#) on page 14

13.3 Storage

Put plastic end caps on the RF connectors to protect them from damage. Protect the power sensor against dust.

Ensure that the environmental conditions, e.g. temperature range and climatic load, meet the values specified in the data sheet.

13.4 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 13-1: 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.

List of commands

*CLS.....	36
*ESE.....	37
*ESR?.....	37
*IDN?.....	37
*IST?.....	37
*OPC.....	37
*OPT?.....	38
*PRE.....	38
*RCL.....	38
*RST.....	38
*SAV.....	38
*SRE.....	39
*STB?.....	39
*TRG.....	39
*TST?.....	39
*WAI.....	39
ABORT.....	42
CALCulate:FEED.....	41
CALibration:DATA.....	108
CALibration:DATA:LENGth?.....	109
CALibration:USER:DATA.....	109
CALibration:USER:DATA:LENGth?.....	109
CALibration:ZERO:FAST:AUTO.....	109
CALibration<Channel>:ZERO:AUTO.....	110
FETCH<Sensor>:ARRAy[:POWer][:AVG]?.....	96
FETCH<Sensor>[:SCALAr][:POWer]:BURSt?.....	97
FETCH<Sensor>[:SCALAr][:POWer]:TSLot?.....	97
FETCH<Sensor>[:SCALAr][:POWer][:AVG]?.....	96
FORMat:BORDER.....	95
FORMat:SREGister.....	95
FORMat[:DATA].....	95
INITiate:ALL.....	43
INITiate:CONTInuous.....	43
INITiate[:IMMediate].....	43
SENSe<Sensor>[:POWer][:AVG:]APERture.....	57
SENSe<Sensor>[:POWer][:AVG:]BUFFer:CLEAr.....	58
SENSe<Sensor>[:POWer][:AVG:]BUFFer:COUNT?.....	58
SENSe<Sensor>[:POWer][:AVG:]BUFFer:DATA?.....	97
SENSe<Sensor>[:POWer][:AVG:]BUFFer:SIZE.....	58
SENSe<Sensor>[:POWer][:AVG:]BUFFer:STATe.....	59
SENSe<Sensor>[:POWer][:AVG:]SMOothing:STATe.....	57
SENSe<Sensor>[:POWer]:BURSt:DTOLerance.....	60
SENSe<Sensor>[:POWer]:BURSt:LENGth?.....	60
SENSe<Sensor>[:POWer]:TSLot[:AVG]:COUNT.....	61
SENSe<Sensor>[:POWer]:TSLot[:AVG]:WIDTh.....	62
SENSe<Sensor>[:POWer]:TSLot[:AVG][:EXCLude]:MID:OFFSet[:TIME].....	62
SENSe<Sensor>[:POWer]:TSLot[:AVG][:EXCLude]:MID:TIME.....	62

SENSe<Sensor>:[POWer:]TSLot[:AVG][:EXCLude]:MID[:STATe].....	62
SENSe<Sensor>:AUXiliary.....	42
SENSe<Sensor>:AVERAge:COUNT.....	75
SENSe<Sensor>:AVERAge:COUNT:AUTO.....	75
SENSe<Sensor>:AVERAge:COUNT:AUTO:MTIME.....	75
SENSe<Sensor>:AVERAge:COUNT:AUTO:NSRatio.....	76
SENSe<Sensor>:AVERAge:COUNT:AUTO:RESolution.....	76
SENSe<Sensor>:AVERAge:COUNT:AUTO:SLOT.....	76
SENSe<Sensor>:AVERAge:COUNT:AUTO:TYPE.....	77
SENSe<Sensor>:AVERAge:RESet.....	77
SENSe<Sensor>:AVERAge:TCONtrol.....	77
SENSe<Sensor>:AVERAge[:STATe].....	78
SENSe<Sensor>:BWIDth:VIDeo.....	63
SENSe<Sensor>:BWIDth:VIDeo:LIST?.....	64
SENSe<Sensor>:CORRection:DCYCle.....	80
SENSe<Sensor>:CORRection:DCYCle:STATe.....	80
SENSe<Sensor>:CORRection:OFFSet.....	81
SENSe<Sensor>:CORRection:OFFSet:STATe.....	81
SENSe<Sensor>:CORRection:SPDevice:LIST?.....	82
SENSe<Sensor>:CORRection:SPDevice:SElect.....	82
SENSe<Sensor>:CORRection:SPDevice:STATe.....	82
SENSe<Sensor>:FREQuency.....	78
SENSe<Sensor>:FUNCTion.....	40
SENSe<Sensor>:SGAMma:CORRection:STATe.....	83
SENSe<Sensor>:SGAMma:MAGNitude.....	83
SENSe<Sensor>:SGAMma:PHASe.....	84
SENSe<Sensor>:STATistics:OFFSet:TIME.....	72
SENSe<Sensor>:STATistics:POWer:AVG?.....	107
SENSe<Sensor>:STATistics:POWer:PEAK:HOLD.....	107
SENSe<Sensor>:STATistics:POWer:PEAK:RESet.....	108
SENSe<Sensor>:STATistics:POWer:PEAK?.....	107
SENSe<Sensor>:STATistics:SCALe:X:MPWidth?.....	73
SENSe<Sensor>:STATistics:SCALe:X:POINts.....	73
SENSe<Sensor>:STATistics:SCALe:X:RANGe.....	74
SENSe<Sensor>:STATistics:SCALe:X:RLEVel.....	74
SENSe<Sensor>:STATistics:TIME.....	73
SENSe<Sensor>:STATistics[:EXCLude]:MID:OFFSet[:TIME].....	72
SENSe<Sensor>:STATistics[:EXCLude]:MID:TIME.....	72
SENSe<Sensor>:TIMing:EXCLude:STARt.....	79
SENSe<Sensor>:TIMing:EXCLude:STOP.....	79
SENSe<Sensor>:TRACe:AVERAge:COUNT.....	64
SENSe<Sensor>:TRACe:AVERAge:TCONtrol.....	64
SENSe<Sensor>:TRACe:AVERAge[:STATe].....	65
SENSe<Sensor>:TRACe:DATA?.....	97
SENSe<Sensor>:TRACe:ESAMpling:AUTO.....	70
SENSe<Sensor>:TRACe:FFT:DATA?.....	99
SENSe<Sensor>:TRACe:FFT:WINDow.....	65
SENSe<Sensor>:TRACe:MEASurement:ALGorithm.....	68
SENSe<Sensor>:TRACe:MEASurement:DEFine:DURation:REFerence.....	69
SENSe<Sensor>:TRACe:MEASurement:DEFine:TRANsition:HREFerence.....	69

SENSe<Sensor>:TRACe:MEASurement:DEFine:TRANSition:LREFerence.....	69
SENSe<Sensor>:TRACe:MEASurement:OFFSet:TIME.....	101
SENSe<Sensor>:TRACe:MEASurement:POWer:AVG?.....	104
SENSe<Sensor>:TRACe:MEASurement:POWer:HREFerence?.....	105
SENSe<Sensor>:TRACe:MEASurement:POWer:LREFerence?.....	105
SENSe<Sensor>:TRACe:MEASurement:POWer:MAX?.....	105
SENSe<Sensor>:TRACe:MEASurement:POWer:MIN?.....	105
SENSe<Sensor>:TRACe:MEASurement:POWer:PULSe:BASE?.....	105
SENSe<Sensor>:TRACe:MEASurement:POWer:PULSe:DROOp?.....	106
SENSe<Sensor>:TRACe:MEASurement:POWer:PULSe:TOP?.....	106
SENSe<Sensor>:TRACe:MEASurement:POWer:REFerence?.....	106
SENSe<Sensor>:TRACe:MEASurement:PULSe:DCYCLE?.....	102
SENSe<Sensor>:TRACe:MEASurement:PULSe:DURation?.....	102
SENSe<Sensor>:TRACe:MEASurement:PULSe:PERiod?.....	102
SENSe<Sensor>:TRACe:MEASurement:PULSe:SEPAration?.....	103
SENSe<Sensor>:TRACe:MEASurement:STATe.....	67
SENSe<Sensor>:TRACe:MEASurement:TIME.....	68
SENSe<Sensor>:TRACe:MEASurement:TRANSition:ESAMpling:AUTO.....	70
SENSe<Sensor>:TRACe:MEASurement:TRANSition:NEGative:DURation?.....	103
SENSe<Sensor>:TRACe:MEASurement:TRANSition:NEGative:OCCurrence?.....	103
SENSe<Sensor>:TRACe:MEASurement:TRANSition:NEGative:OVERshoot?.....	106
SENSe<Sensor>:TRACe:MEASurement:TRANSition:POSitive:DURation?.....	103
SENSe<Sensor>:TRACe:MEASurement:TRANSition:POSitive:OCCurrence?.....	104
SENSe<Sensor>:TRACe:MEASurement:TRANSition:POSitive:OVERshoot?.....	106
SENSe<Sensor>:TRACe:MEASurement:TRANSition:SPERiod?.....	104
SENSe<Sensor>:TRACe:MPWidth?.....	66
SENSe<Sensor>:TRACe:OFFSet:TIME.....	66
SENSe<Sensor>:TRACe:POINts.....	66
SENSe<Sensor>:TRACe:POINts:FPGA?.....	66
SENSe<Sensor>:TRACe:TIME.....	66
SENSe<Sensor>:TRACe:TIME:FPGA?.....	67
STATus:DEVice:CONDition?.....	119
STATus:DEVice:ENABle.....	120
STATus:DEVice:NTRansition.....	120
STATus:DEVice:PTRansition.....	120
STATus:DEVice[:EVENT]?.....	119
STATus:OPERation:CALibrating:CONDition?.....	119
STATus:OPERation:CALibrating:ENABle.....	120
STATus:OPERation:CALibrating:NTRansition.....	120
STATus:OPERation:CALibrating:PTRansition.....	120
STATus:OPERation:CALibrating[:SUMMARY][:EVENT]?.....	119
STATus:OPERation:CONDition?.....	119
STATus:OPERation:ENABle.....	120
STATus:OPERation:MEASuring:CONDition?.....	119
STATus:OPERation:MEASuring:ENABle.....	120
STATus:OPERation:MEASuring:NTRansition.....	120
STATus:OPERation:MEASuring:PTRansition.....	120
STATus:OPERation:MEASuring[:SUMMARY][:EVENT]?.....	119
STATus:OPERation:NTRansition.....	120
STATus:OPERation:PTRansition.....	120

STATus:OPERation:SENSe:CONDition?	119
STATus:OPERation:SENSe:ENABle	120
STATus:OPERation:SENSe:NTRansition	120
STATus:OPERation:SENSe:PTRansition	120
STATus:OPERation:SENSe[:SUMMary][:EVENT]?	119
STATus:OPERation:TRIGger:CONDition?	119
STATus:OPERation:TRIGger:ENABle	120
STATus:OPERation:TRIGger:NTRansition	120
STATus:OPERation:TRIGger:PTRansition	120
STATus:OPERation:TRIGger[:SUMMary][:EVENT]?	119
STATus:OPERation[:EVENT]?	119
STATus:PRESet	119
STATus:QUEStionable:CALibration:CONDition?	119
STATus:QUEStionable:CALibration:ENABle	120
STATus:QUEStionable:CALibration:NTRansition	120
STATus:QUEStionable:CALibration:PTRansition	120
STATus:QUEStionable:CALibration[:SUMMary][:EVENT]?	119
STATus:QUEStionable:CONDition?	119
STATus:QUEStionable:ENABle	120
STATus:QUEStionable:NTRansition	120
STATus:QUEStionable:POWer:CONDition?	119
STATus:QUEStionable:POWer:ENABle	120
STATus:QUEStionable:POWer:NTRansition	120
STATus:QUEStionable:POWer:PTRansition	120
STATus:QUEStionable:POWer[:SUMMary][:EVENT]?	119
STATus:QUEStionable:PTRansition	120
STATus:QUEStionable[:EVENT]?	119
STATus:QUEue[:NEXT]?	119
SYSTem:DFPPrint<Channel>?	111
SYSTem:ERRor:ALL?	112
SYSTem:ERRor:CODE:ALL?	112
SYSTem:ERRor:CODE[:NEXT]?	112
SYSTem:ERRor:COUNT?	112
SYSTem:ERRor[:NEXT]?	113
SYSTem:FWUPdate	113
SYSTem:FWUPdate:STATus?	113
SYSTem:HELP:HEADers?	114
SYSTem:INFO?	114
SYSTem:INITialize	114
SYSTem:LANGuage	114
SYSTem:LED:COLor	115
SYSTem:LED:MODE	115
SYSTem:MINPower?	115
SYSTem:PARAmeters:DELTA?	116
SYSTem:PARAmeters?	116
SYSTem:PRESet	116
SYSTem:REBoot	116
SYSTem:REStart	117
SYSTem:RUTime	117
SYSTem:SERRor?	117

SYSTem:SUTime.....	117
SYSTem:TLEVels?.....	117
SYSTem:TRANsaction:BEGiN.....	117
SYSTem:TRANsaction:END.....	118
SYSTem:VERSiOn?.....	118
SYSTem[:SENSor]:NAME.....	118
TEST:SENSor?.....	110
TRIGger:ATRigger:DELaY.....	51
TRIGger:ATRigger:EXECuted?.....	51
TRIGger:ATRigger[:STATe].....	51
TRIGger:COUnT.....	52
TRIGger:DELaY.....	52
TRIGger:DTIME.....	52
TRIGger:EXTErnal<2...2>:IMPedance.....	53
TRIGger:HOLDoff.....	53
TRIGger:HYSteresis.....	53
TRIGger:IMMediate.....	53
TRIGger:LEVel.....	54
TRIGger:LEVel:UNIT.....	54
TRIGger:SEnDer:PORT.....	54
TRIGger:SEnDer:STATe.....	55
TRIGger:SLOPe.....	55
TRIGger:SOURce.....	55
TRIGger:SYNC:STATe.....	56
UNIT:POWer.....	94

Index

A

Application cards	11
Application notes	11
Auto averaging	74

B

Basic measurement settings	
Basic configuration	74
Brochures	11
Burst average measurement	59

C

Calibrating	108
Checking	12
Cleaning	144
Common command	36
Computer	
Connection types	15
Simple USB connection	15
USB sensor hub	16
Connecting	
Base unit	18
RF	13
Supported R&S instrument	18
To a computer	15
USB host	14
Connector	
Host interface	21
Overview	20
RF	20
Trigger I/O	21
Continuous average measurement	56
Buffered measurement	121
Controlling host	15
Correction	80
Duty cycle	80
Offset	80
S-gamma	82
S-parameter	81
Customer support	141

D

Data sheets	11
Device status register	129
Diagnostics	
Selftest	140
Disconnecting	
Host interface	15
RF	14
Documentation overview	10
Download	
R&S NRP Toolkit	22
Dropout time	
Trigger	45
DUT	
Connecting	13
Disconnecting	13
Duty cycle	80

E

Electromagnetic interference (EMI)	13
Electrostatic discharge (ESD)	13
Enable register	133
ESE	133
ESR	133
Exclusion time	78

F

Firmware update	24, 29
Firmware Update for NRP Family	29
Preparing	29
Problems	140
Remote control	31
Troubleshooting	31
Frequency	78

G

Getting started	10
-----------------------	----

H

Hold-off time	
Trigger	46

I

Installation	
Firmware	29
Instrument security procedures	10
Intended use	7
Interface	
Overview	20

K

Key facts	11
Keywords	
optional	35

L

Labels	8
LAN connection	
Configuring	24
LED	
Status	21
Low-level communication program	24

M

Maintenance	143
Cleaning	144
Regular checks	143
Measured values	
Remote control	41
Measurement	
Burst average	59
Continuous average	56
Ending	42
Preparing	40

Pulse analysis	67
Starting	42
Statistics	70
Timeslot	60
Trace	62
Measurement control	42
Continuous average mode	47
Trace mode	49
Trigger	43
Measurement mode	
Configuring	56
Remote control	40
Measurement results	
Controlling	46
N	
Notation	
Remote control	35
O	
Offset	80
Open source acknowledgment (OSA)	11
Operating concepts	22
R&S NRP Toolkit	22
R&S NRPV	25
R&S NRX	27
R&S Power Viewer	26
Remote control	24
Operating conditions	12
Operation status registers	134
P	
Power supply	14
Preparing for use	12
Programming example	121
Buffered measurement	121
Simplest measurement	121
Trace measurement	123, 124
Pulse analysis measurement	67
Q	
Questionable status registers	131
R	
R&S NRP Toolkit	22
Communication program	24
Components for Windows-based systems	23
Firmware update	24
Installation under Windows	23
Network configuration	24
S-parameter update	24
System requirements	22
Uncertainty calculator	24
Versions	22
Versions of R&S NRP software	24
R&S NRP-Z5	16
R&S NRPV	25
R&S NRX	27
R&S Power Viewer	26
Regular checks	143
Regulatory information	7
Release notes	11
Remote control	
Basic measurement settings	74
Basics	126
Burst average measurement	59
Common commands	36
Continuous average measurement	56
Conventions	34
Ending measurement	42
Interfaces	126
Measurement mode	56
Notation	35
Preparing for measurement	40
Protocols	126
Pulse analysis measurement	67
Results	94
Starting measurement	42
Statistics measurement	70
Timeslot measurement	60
Trace measurement	62
Trigger configuration	51
USB interface	126
Replacement	
Power sensor	33
Results	
Configuring	94
Format	95
Pulse analysis	100
Retrieving	96
Statistics	107
Unit	94
RF connector	13, 20
RSU file	29
S	
S-gamma	82
S-parameter correction	81, 84
S-parameter update	24
S-Parameters program	84
Configuration	88
S2P measurement data files	92
Uncertainty data files	93
User interface	85
Safety information	7, 12
Safety instructions	7, 10
Security procedures	10
Selftest	110, 140
Serial number	8
Sources	
Trigger	44
Special characters	
SCPI	36
Standard event status register	133
Statistics measurement	70
Status information	140
Status LED	21
Status register	118
CONDition part	119
Device status register	129
ENABLE part	120
Enable register	133
EVENT part	119
General commands	119
Hierarchy	127
NTRansition part	120
Operation calibrating status register	135
Operation measuring status register	136

Operation sense status register	138
Operation status register	134
Operation trigger status register	137
PTRansition part	120
Questionable calibration status register	132
Questionable power status register	132
Questionable status register	131
Standard event status register	133
Suffixes	35
System configuration	111

T

Target audience	7
Termination control	46
Test setup	13
Timeslot measurement	60
Trace measurement	62, 123
Synchronized	124
Transporting	142
Trigger	43
Configuring	51
Dropout time	45
Event	44
Hold-off time	46
Sources	44
States	44
Trigger I/O connector	21
Troubleshooting	140
Firmware update	140
Selftest	140
Status information	140
Tutorials	10

U

Uncertainty calculator	24
Units	34
Unpacking	12
Update file	29
USB connection	15
USB interface	
Remote control	126
USB product ID	127
USB resource string	127
USB sensor hub	16
User manual	10

V

Versions of R&S NRP software	24
------------------------------------	----

W

Warning message	9
White papers	11

Z

Zeroing	108
---------------	-----