R&S®FPS-K18 and -K18D
Power Amplifier Measurements
User Manual

1177610002
Version 05

ROHDE&SCHWARZ
Make ideas real

This manual applies to the following R&S®FPS models with firmware version 1.70 and higher:

- R&S®FPS4 (1319.2008K04)
- R&S®FPS7 (1319.2008K07)
- R&S®FPS13 (1319.2008K13)
- R&S®FPS30 (1319.2008K30)
- R&S®FPS40 (1319.2008K40)

The following firmware options are described:

- R&S FPS-K18 (1321.4662.02)
- R&S FPS-K18D (1321.4956.02)

Mühldorfstr. 15, 81671 München, Germany
Phone: +49 89 41 29 - 0
Email: info@rohde-schwarz.com
Internet: www.rohde-schwarz.com

1177.6100.02 | Version 05 | R&S®FPS-K18 and -K18D

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

Contents

1 Preface

1.1 About this Manual

This Amplifier Measurements User Manual provides all the information **specific to the application**. All general instrument functions and settings common to all applications and operating modes are described in the main R&S FPS User Manual.

The main focus in this manual is on the amplifier measurement results and the tasks required to obtain them. The following topics are included:

- **Welcome to the Amplifier application**
  Introduction to and getting familiar with the application
- **Measurements and result displays**
  Details on supported measurements and their result types
- **Configuration and analysis**
  A concise description of all functions and settings available to configure and analyze amplifier measurements with their corresponding remote control command
- **Remote commands for amplifier measurements**
  Remote commands required to configure and perform amplifier measurements in a remote environment, sorted by tasks
  (Commands required to set up the environment or to perform common tasks on the instrument are provided in the main R&S FPS User Manual)
- **List of remote commands**
  Alpahabetical list of all remote commands described in the manual
- **Index**

1.2 Typographical Conventions

The following text markers are used throughout this documentation:

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

2 Welcome to the Amplifier Measurement Application

The R&S FPS-K18 is a firmware application that adds functionality to measure the efficiency of amplifiers with the R&S FPS signal analyzer. You extend the amplifier application with the R&S FPS-K18D, which adds direct DPD functionality.

This user manual contains a description of the functionality that the application provides, including remote control operation.

Functions that are not discussed in this manual are the same as in the base unit and are described in the R&S FPS user manual. The latest versions of the manuals are available for download at the product homepage.

http://www.rohde-schwarz.com/product/FPS.html.

Installation

Find detailed installing instructions in the getting started or the release notes of the R&S FPS.

2.1 Starting the Application

The amplifier measurement application adds a new type of measurement to the R&S FPS.

To activate the amplifier application

1. Press the [MODE] key on the front panel of the R&S FPS.

   A dialog box opens that contains all operating modes and applications currently available on your R&S FPS.

2. Select the "Amplifier" item.

   The R&S FPS opens a new measurement channel for the amplifier application. All settings specific to amplifier measurements are in their default state.

2.2 Understanding the Display Information

The following figure shows the display as it looks for amplifier measurements. All different information areas are labeled. They are explained in more detail in the following sections.

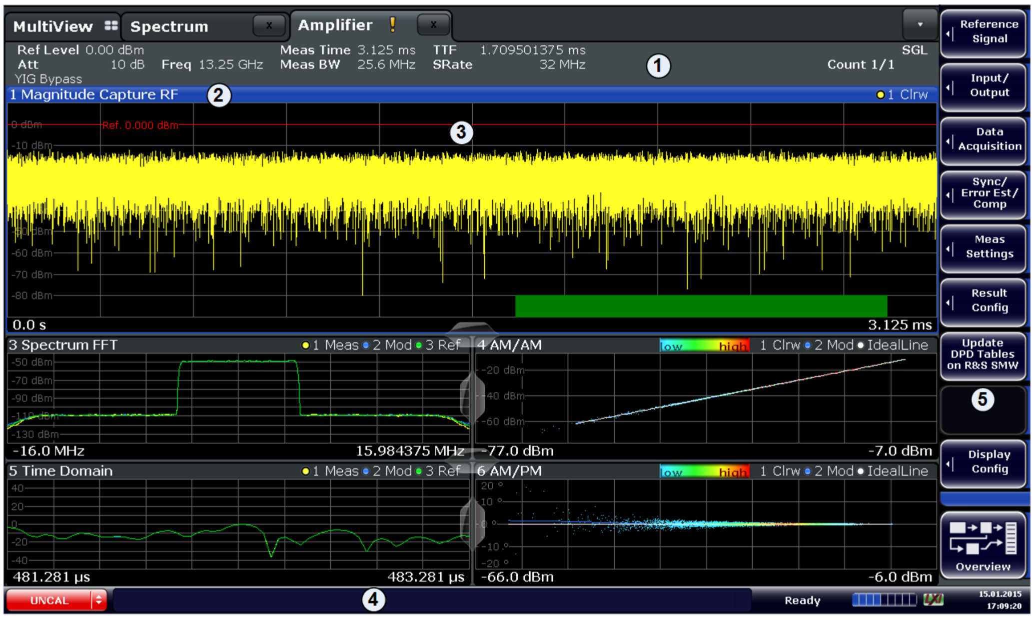


Figure 2-1: Screen layout of the amplifier measurement application

1 = Toolbar
2 = Channel bar
3 = Result display
4 = Status bar
5 = Softkey bar

For a description of the elements not described below, refer to the getting started of the R&S FPS.

Channel bar information

The channel bar contains information about the current measurement setup, progress and results.

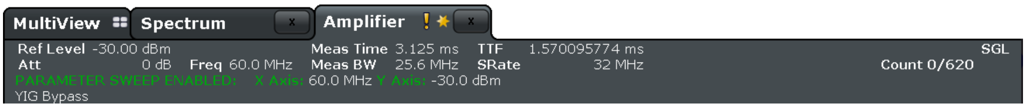


Figure 2-2: Channel bar of the amplifier application

Ref Level	Current reference level of the analyzer.
Att	Current attenuation of the analyzer.
Freq	Frequency the signal is transmitted on.
Meas Time	Length of the signal capture.

Meas BW	Bandwidth with which the signal is recorded.
TTF	Time difference between the trigger event and the first sample of the reference signal (= beginning of a frame).
SRate	Sample rate with which the signal is recorded.
SGL	Indicates that single sweep mode is active.
Count	The current signal count for measurement tasks that involve a specific number of subsequent sweeps (for example the parameter sweep).
X Axis	X-axis value that is currently measured.
Y Axis	Y-axis value that is currently measured.

Window title bar information

For each diagram, the header provides the following information:

Figure 2-3: Window title bar information of the amplifier application

1 = Window number
2 = Window type
3 = Trace color and number
4 = Trace mode
Blue color = Window is selected

Status bar information

Global instrument settings, the instrument status and any irregularities are indicated in the status bar beneath the diagram. Furthermore, the progress of the current operation is displayed in the status bar.

3 Measurements and Result Displays

Note that you can use the R&S FPS-K18 with the sequencer that is available with the R&S FPS. The functionality is the same as in the spectrum application. Refer to the R&S FPS user manual for more information.

Adjacent Channel Leakage Error (ACLR)

The "ACLR" result display shows the power characteristics of the transmission (Tx) channel and its neighboring channel(s).

The ACLR measurement in the R&S FPS-K18 is a measurement based on I/Q data. Thus, its results are calculated by the same I/Q data as the rest of the results (like the EVM). Note that the supported channel bandwidth is limited by the I/Q bandwidth of the analyzer you are using.

The results are provided in numerical form in a table. The table is made up out of two parts, one part containing the characteristics of the Tx channel, the other containing those of the neighboring channels.

2 ACLR

Channel	Bandw...	Offset	Power		
Tx1 (Re...	9.015 M...		10.902 ...		
Tx Total			10.902 ...		
Channel	**Bandw...**	**Offset**	**Lower**	**Upper**	**Balanced**
Adj	9.015 M...	10.000 ...	-2.628 ...	-3.306 ...	0.677 dB
Alt 1	9.015 M...	20.000 ...	-36.034...	-35.311...	-0.723 dB

The table contains the following information.

- **Channel**
  Shows the type of channel.
- **Bandwidth**
  Shows the channel's bandwidth.
- **Offset** (neighboring channels only)
  Shows the frequency offset between the center frequency of the adjacent (or alternate) channel and the center frequency of the transmission channel.
- **Power**
  Shows the power of the transmission channel, or the power of the upper / lower neighboring channel.
  The result is calculated over the complete capture buffer, not just the evaluation range.
- **Balanced**
  Shows the difference between the lower and upper adjacent channel power ("Lower Channel" - "Upper Channel").

For more information on configuring the ACP measurement, see Chapter 4.15, "Configuring Adjacent Channel Leakage Error (ACLR) Measurements", on page 86.

Remote command:
Selection: `LAY:ADD? '1',LEFT,ACP`
Result query: `CALCulate<n>:MARKer<m>:FUNCtion:POWer:RESult?` on page 270

AM/AM

The "AM/AM" result display shows nonlinear effects of the DUT. It shows the amplitude at the DUT input against the amplitude at the DUT output.

The ideal AM/AM curve would be a straight line at 45°. However, nonlinear effects result in a measurement curve that does not follow the ideal curve. When you drive the amplifier into saturation, the curve typically flattens at high input levels.

The width of the AM/AM trace is an indicator of memory effects: the larger the width of the trace, the more memory effects occur. The AM/AM Curve Width is shown in the numerical Result Summary.

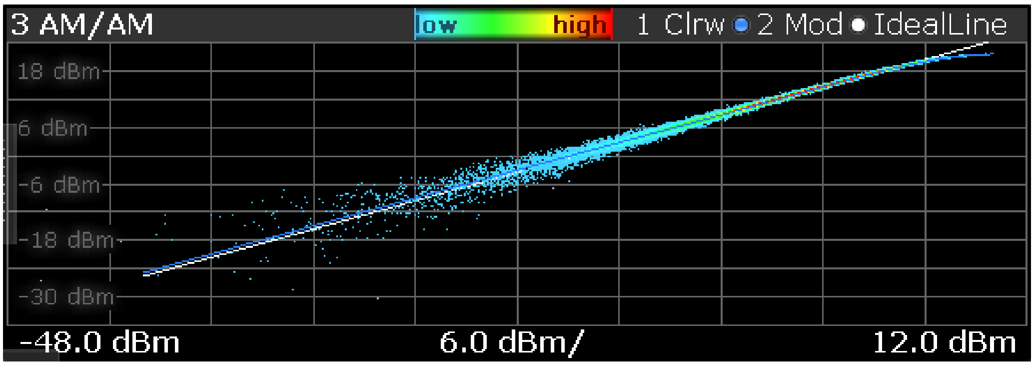
Both axes show the power of the signal in dBm.

You can analyze the AM/AM characteristics of the measured signal and the modeled signal.

- Measured signal
  Shows the AM/AM characteristics of the DUT.
  The software uses the reference signal in combination with the synchronized measurement signal to calculate a software model that describes the characteristics of the device under test.
  The measured signal is represented by a colored cloud of values. The cloud is based on the recorded samples. If samples have the same values (and would thus be superimposed), colors represent the statistical frequency with which a certain input / output level combination occurs. Blue pixels represent low statistical frequencies, red pixels high statistical frequencies. A color map is provided within the result display.
- Modeled signal

Shows the AM/AM characteristics of the model that has been calculated. The modeled signal is calculated by applying the DUT model to the reference signal.
When the model matches the characteristics of the DUT, the characteristics of the model signal are the same as those of the measured signal (minus noise).
The modeled signal is represented by a line trace.
When system modeling has been turned off, this trace is not displayed.

All traces include the digital predistortion, when you have turned on that feature.



Remote command:
Selection: `LAY:ADD? '1',LEFT,AMAM`
Result query: `TRACe<n>[:DATA]?` on page 129

AM/PM

The "AM/PM" result display shows nonlinear effects of the DUT. It shows the phase difference between DUT input and output for each sample of the synchronized measurement signal.

The ideal AM/PM curve would be a straight line at 0°. However, nonlinear effects result in a measurement curve that does not follow the ideal curve. Typically, the curve drifts from a zero phase shift, especially at high power levels when you drive the amplifier into saturation.

The width of the AM/PM trace is an indicator of memory effects: the larger the width of the trace, the more memory effects occur. The AM/PM curve width is shown in the numerical Result Summary.

The x-axis shows the levels of all samples of the reference signal (input power) or the measurement signal (output power) in dBm. You can select the reference of the x-axis (input or output power) in the "Result Configuration" dialog box.

The y-axis shows the phase of the signal for the corresponding power level. The unit is either rad or degree, depending on your phase unit selection in the "Result Configuration" dialog box.

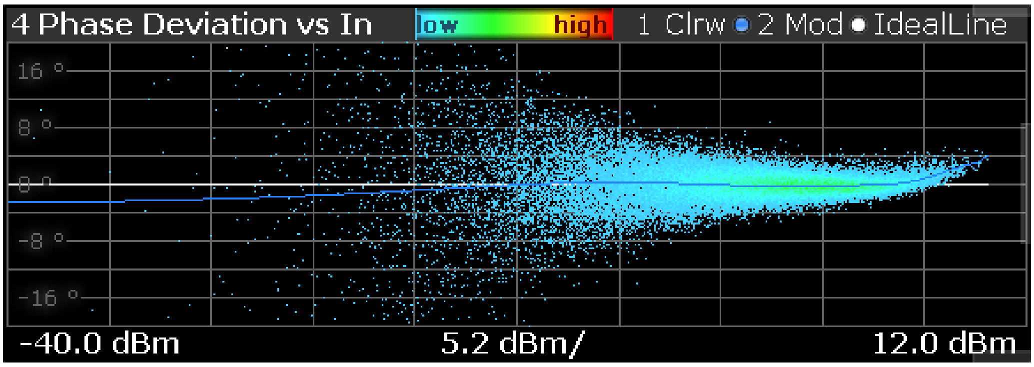
You can analyze the AM/PM characteristics of the real DUT or of the modeled DUT.

- Measured signal
  Shows the AM/PM characteristics of the DUT.
  The software uses the reference signal together with the synchronized measurement signal to calculate a software model that describes the characteristics of the device under test.
  The measured signal is represented by a colored cloud of values. The cloud is based on the recorded samples. If samples have the same values (and would thus

be superimposed), colors represent the statistical frequency with which a certain input / output level combination occurs. A color map is provided within the result display.

- Modeled signal
  Shows the AM/PM characteristics of the model that has been calculated. The modeled signal is calculated by applying the DUT model to the reference signal.
  When the model matches the characteristics of the DUT, the characteristics of the modeled signal are the same as those of the measured signal (minus noise).
  The modeled signal is represented by a line trace.
  When system modeling has been turned off, this trace is not displayed.

All traces include the digital predistortion, when you have turned on that feature.



Remote command:
Selection: `LAY:ADD? '1',LEFT,AMPM`
Result query: `TRACe<n>[:DATA]?` on page 129

EVM vs Power

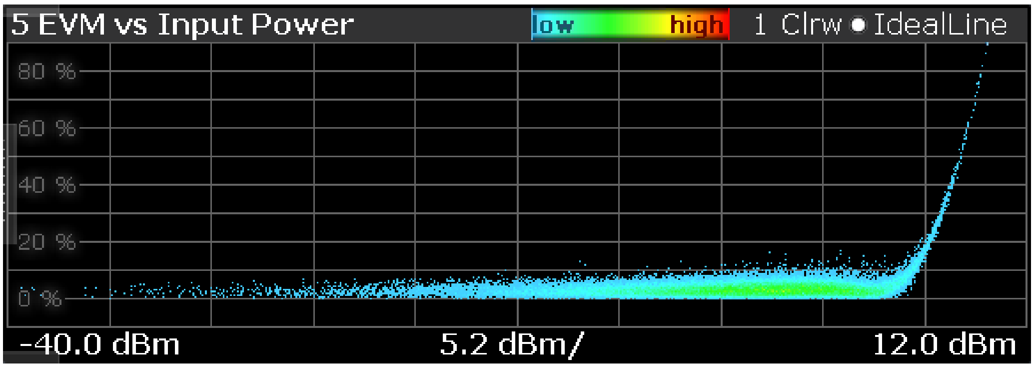
The "EVM vs Power" result display shows the EVM against the measured power values.

The ideal EVM vs power curve would be a straight line at 0 %. However, among other effects such as noise, nonlinear effects of the DUT cause an increase of the EVM. Nonlinear effects usually occur on high power levels that drive the power amplifier into saturation.

The x-axis shows the levels of all samples of the reference signal (input power) or the measurement signal (output power) in dBm. You can select the reference of the x-axis (input or output power) in the "Result Configuration" dialog box.

The y-axis shows the EVM of the signal for the corresponding power level in %.

All traces include the digital predistortion, when you have turned on that feature.



Remote command:
Selection: LAY:ADD? '1',LEFT,AMEV
Result query: TRACe<n>[:DATA]? on page 129

Error Vector Spectrum

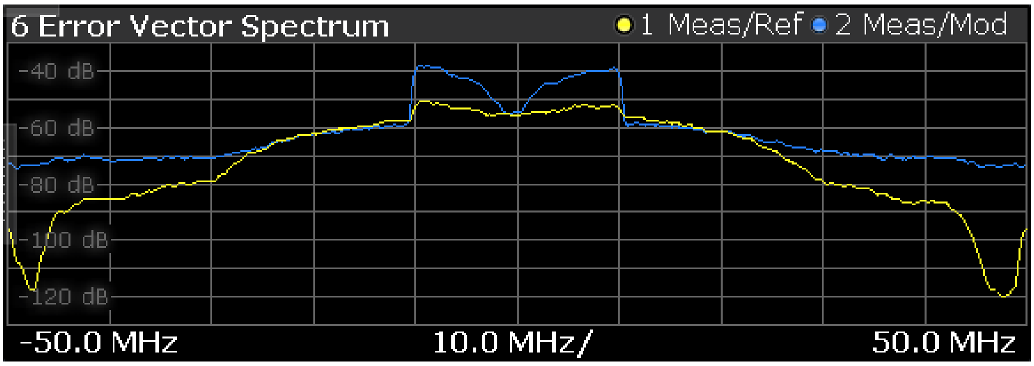
The "Error Vector Spectrum" result display shows the error vector (EV) signal in the spectrum around the center frequency.

The EV is a measure of the modulation accuracy. It compares two signals and shows the distance of the measured constellation points and the ideal constellation points.

The unit is dB.

You can compare the measured signal against the reference signal and against the modeled signal.

- Measured signal against reference signal
  Trace 1 compares measured signal and the reference signal.
  To get useful results, the calculated linear gain is compensated to match both signals.
  Depending on the DUT, noise and nonlinear effects may have been added to the measurement signal. These effects are visualized by this trace.
- Measured signal against modeled signal
  Trace 2 compares measured signal and the modeled signal.
  The EVM between the measured and modeled signal indicates the quality of the DUT modeling. If the model matches the DUT behavior, the modeling error is zero (or is merely influenced by noise).
  This result display shows changes in the model and its parameters and thus allows you to optimize the modeling.
  When system modeling has been turned off, this trace is not displayed.



Remote command:
Selection: LAY:ADD? '1',LEFT,SEVM
Result query: TRACe<n>[:DATA]? on page 129

Gain Compression

The "Gain Compression" result display shows the gain and error effects of the DUT against the DUT input or output power.

The gain is the ratio of the input and output power of the DUT.

The x-axis shows the levels of all samples of the reference signal (input power) or the measurement signal (output power) in dBm. You can select the reference of the x-axis (input or output power) in the "Result Configuration" dialog box.

The y-axis shows the gain in dB.

The ideal gain compression curve would be a straight horizontal line. However, nonlinear effects result in a measurement curve that does not follow the ideal curve. In addition, the curve widens at very low input levels due to noise influence.

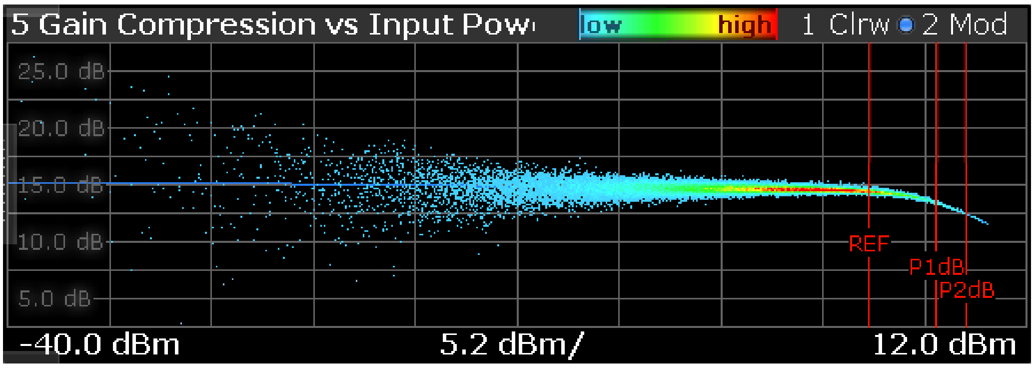
The width of the gain compression trace is an indicator of memory effects: the larger the width of the trace, the more memory effects occur.

You can analyze the gain characteristics of the measured signal and the modeled signal.

- Measured signal
  Shows the gain characteristics of the DUT.
  The software uses the reference signal in combination with the synchronized measurement signal to calculate a software model that describes the characteristics of the device under test.
  The measured gain is represented by a colored cloud of values. The cloud is based on the recorded samples. If samples have the same values (and would thus be superimposed), colors represent the statistical frequency with which a certain input / output level combination occurs. Blue pixels represent low statistical frequencies, red pixels high statistical frequencies. A color map is provided within the result display.
- Modeled signal
  Shows the gain characteristics of the model that has been calculated. The modeled signal is calculated by applying the DUT model to the reference signal.
  When the model matches the characteristics of the DUT, the characteristics of the model signal are the same as those of the measured signal (minus noise).
  The modeled signal is represented by a line trace.
  When system modeling has been turned off, this trace is not displayed.

In addition, one or more horizontal lines can appear in the result display.

- One line to indicate each compression point (1 dB, 2 dB and 3 dB).
- One line to indicate the reference point (0 dB compression) that the compression points refer to.



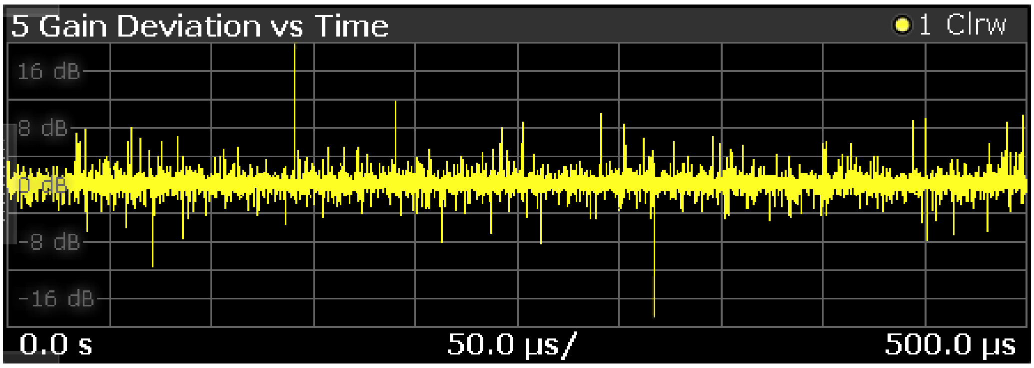
Remote command:
Selection: LAY:ADD? '1',LEFT,GC
Result query: TRACe<n>[:DATA]? on page 129

Gain Deviation vs Time
The "Gain Deviation vs Time" result display shows the deviation of each measured signal sample from the average gain of the measured signal.

The x-axis shows the time in seconds. The y-axis shows the gain deviation in dB.

The displayed results are based on the synchronized measurement data (represented by the green bar in the capture buffer).

Note that the result query and trace export only work for unencrypted reference signal waveform files.



Remote command:
Selection: LAY:ADD? '1',LEFT,GDVT
Result query: TRACe<n>[:DATA]? on page 129

Magnitude Capture
The "Magnitude Capture" result display contains the raw data that has been recorded and thus represents the characteristics of the DUT.

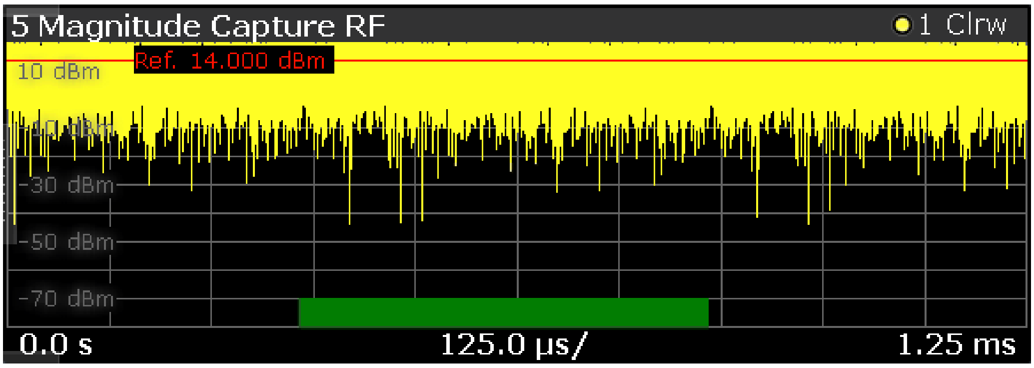
The capture buffer shows the signal level over time. The unit is either dBm.

The raw data is source for all further evaluations. You can also use the data in the capture buffer to identify the causes for possible unexpected results.

When you synchronize the reference signal and the measured signal, the synchronized area is indicated by a horizontal green bar on the bottom of the diagram.

The current reference level is indicated by a red horizontal line.

The green bar at the bottom shows the current frame. In I/Q averaging mode, the average value is shown. In trace statistics mode, multiple values are possible. The currently selected value is symbolized by a blue bar.



Remote command:
Selection (RF): `LAY:ADD? '1',LEFT,RFM`
Result query: `TRACe<n>[:DATA]?` on page 129

Parameter Sweep
The "Parameter Sweep" result display is a result display that shows a result of the DUT (for example the EVM) against two (custom) measurement parameters. The results of this measurement are displayed in graphical and numerical form.

The parameter sweep is a good way, for example, to find the location of the ideal delay time of the RF signal and the envelope signal if you are measuring an amplifier that supports envelope tracking. You can also use the parameter sweep to determine the characteristics and behavior of an amplifier over different frequencies and levels.

For more information about supported parameters and how to set them up see "Selecting the data to be evaluated during the parameter sweep" on page 90.

Parameter Sweep: Diagram ← Parameter Sweep
The parameter sweep diagram is a graphical representation of the parameter sweep results. The results are either represented as a two-dimensional trace or as a three-dimensional trace, depending on whether you are performing a parameter sweep with one or two parameters.

In a two-dimensional diagram, the y-axis always shows the result. The displayed result depends on the result type you have selected. The information displayed on the x-axis depends on the parameter you have selected for evaluation (for example the EVM over a given frequency range). Values between measurement point are interpolated. Basically, you can interpret the two-dimensional diagram as follows (example): "at a frequency of x Hz, the EVM has a value of y."

In a three-dimensional diagram, the z-axis always shows the result. The information on the other two axes is arbitrary and depends on the parameters you have selected for evaluation. For a better readability, the result values in the three-dimensional diagram are represented by a colored trace: low values have a blue color, while high values have a red color. Values between measurement point are interpolated. Basically, you can interpret the three-dimensional diagram as follows (example): "at a frequency of x Hz and a level of y, the EVM has a value of z."

Parameter Sweep: Table ← Parameter Sweep

The parameter sweep table shows the minimum and maximum results for all available result types in numerical form. For each result type, the location where the minimum and maximum result has occurred is displayed.

Example:

5 Parameter Sweep Table

Result		Value	Frequency	Power
EVM	Min	0.878 %	1.23 GHz	-30.0 dBm
	Max	2.095 %	1.3 GHz	-30.0 dBm
ACLR Tx	Min	-20.460 d...	1.28 GHz	-30.0 dBm
	Max	-18.983 d...	1.01 GHz	-30.0 dBm
ACLR Adj 1 Lower	Min	-3.289 dBc	1.3 GHz	-30.0 dBm

A minimum EVM of 0.244 % and a maximum EVM of 0.246 % has been measured (first and second row). The minimum EVM has been measured at a frequency of 30 MHz and an output power of 0 dBm. The maximum EVM has been measured at a frequency of 10 MHz and an output power of 0 dBm.

The following result types are evaluated in the parameter sweep.

Result	Description
EVM	Error vector magnitude between synchronized reference and measurement signal.
ACLR	Power of the transmission channel.
ACLR Adj Upper / Lower	Power of the adjacent channels (upper and lower).
ACLR Balanced (Adj, Alt1 and Alt2)	Difference between the lower and upper adjacent channel power
RMS Power	RMS signal power at the DUT output.
Gain	Gain of the DUT.
Crest Factor Out	Crest factor of the signal at the DUT output. The crest factor is the ratio of the RMS and peak power.
Curve Width (AM/AM, AM/PM)	Spread of the samples in the AM/AM (or AM/PM) result display compared to the ideal AM/AM (or AM/PM) curve.
Power Out	Signal power at the DUT output.

Result	Description
Compression Point (1 dB / 2 dB / 3 dB)	Input power where the gain deviates by 1 dB, 2 dB or 3 dB from a reference gain (see "Configuring compression point calculation" on page 85).
Bal ACLR Magnitude	Shows the difference between the lower and upper adjacent channel power.

Remote command:
Chapter 6.5.3.3, "Retrieving Results of the Parameter Sweep Table", on page 143

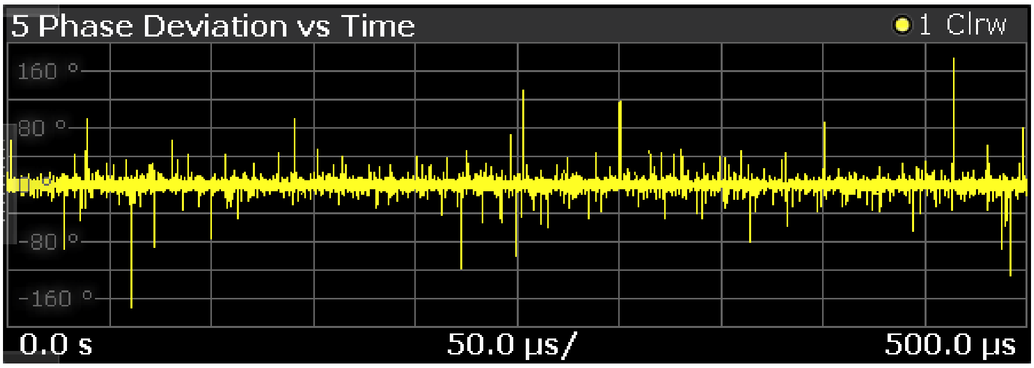
Phase Deviation vs Time

The "Phase Deviation vs Time" result display shows the phase deviation of the measured signal compared to the reference signal over time.

The x-axis shows the time in seconds. The y-axis shows the phase deviation in degree.

The displayed results are based on the synchronized measurement data (represented by the green bar in the capture buffer).

Note that the result query and trace export only work for unencrypted reference signal waveform files.



Remote command:
Selection: LAY:ADD? '1',LEFT,PDVT
Result query: TRACe<n>[:DATA]? on page 129

Raw EVM

The "Raw EVM" result display shows the error vector magnitude of the signal over time.

The EVM is a measure of the modulation accuracy. It compares two signals and shows the distance of the measured constellation points and the ideal constellation points.

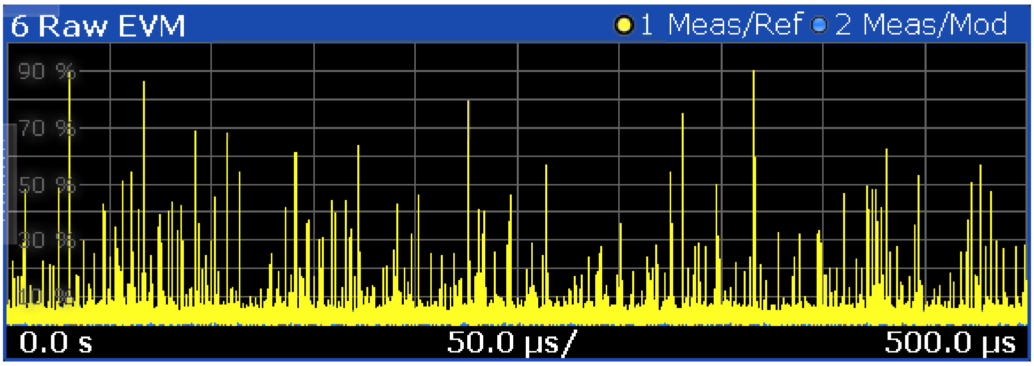
You can compare the measured signal against the reference signal and against the modeled signal.

- Measured signal against reference signal
  Trace 1 compares the measured signal and the reference signal.
  To get useful results, the calculated linear gain is compensated to match both signals.

Depending on the DUT, noise and nonlinear effects may have been added to the measurement signal. These effects are visualized by this trace.

- Measured signal against modeled signal
  Trace 2 compares the measured signal and the modeled signal.
  The EVM between the measured and modeled signal indicates the quality of the DUT modeling. If the model matches the DUT behavior, the modeling error is zero (or is merely influenced by noise).
  This result display shows changes in the model and its parameters and thus allows you to optimize the modeling.
  When system modeling has been turned off, this trace is not displayed.

Note that the raw EVM is calculated for each sample that has been recorded. Thus, the raw EVM can differ from EVM values that are calculated according to a specific mobile communication standard that apply special rules to calculate the EVM, for example LTE.



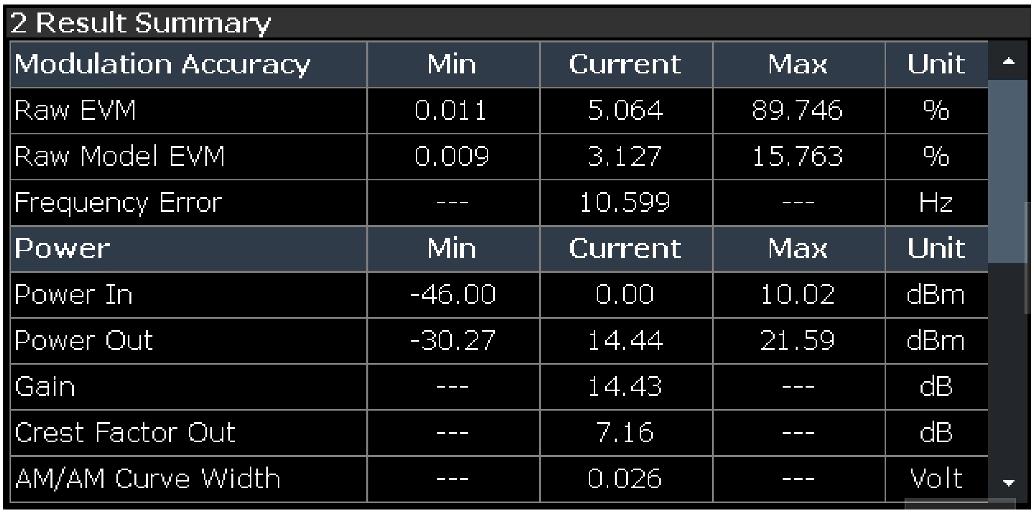
Remote command:
Selection: LAY:ADD? '1',LEFT,REVM
Result query: TRACe<n>[:DATA]? on page 129

Numeric Result Summary

The "Result Summary" shows various measurement results in numerical form, combined in one table.

The table is split in two parts.

- The first part shows the modulation accuracy
- The second part shows the power characteristics of the RF signal



2 Result Summary

Modulation Accuracy	Min	Current	Max	Unit
Raw EVM	0.011	5.064	89.746	%
Raw Model EVM	0.009	3.127	15.763	%
Frequency Error	---	10.599	---	Hz
Power	**Min**	**Current**	**Max**	**Unit**
Power In	-46.00	0.00	10.02	dBm
Power Out	-30.27	14.44	21.59	dBm
Gain	---	14.43	---	dB
Crest Factor Out	---	7.16	---	dB
AM/AM Curve Width	---	0.026	---	Volt

For each result type, several values are displayed.

- **Current**
  Value measured during the last sweep.
  For measurements that evaluate each captured sample, this value represents the average value over all samples captured in the last sweep.
- **Min**
  For measurements that evaluate each captured sample, this value represents the sample with lowest value captured in the last sweep.
- **Max**
  For measurements that evaluate each captured sample, this value represents the sample with the highest value captured in the last sweep.
- **Unit**
  Unit of the result.

Results that evaluate each captured sample

- Raw EVM and Raw Model EVM
- Power In and Power Out

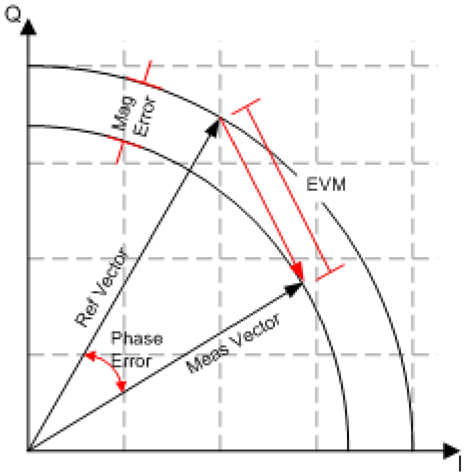
Note: When synchronization has failed or has been turned off, some results may be unavailable.

Remote command:
Selecting the result display: `LAY:ADD? '1',LEFT,RTAB`
Querying results: see Chapter 6.5.3, "Retrieving Numeric Results", on page 131

Results to check modulation accuracy ← Numeric Result Summary

Raw EVM

Error vector magnitude between synchronized reference and measured signal.



FETCh:MACCuracy:REVM:CURRent[:RESult]? on page 136

Raw Model EVM

Error vector magnitude between synchronized measured and model signal.

FETCh:MACCuracy:RMEV:CURRent[:RESult]? on page 136

Frequency Error

Difference of the RF frequency of the reference signal compared to the measured signal.

Note that a frequency error is not available if the frequency error estimation is switched off. See also Chapter 4.10, "Estimating and Compensating Signal Errors", on page 74.

FETCh:MACCuracy:FERRor:CURRent[:RESult]? on page 133

Sample Rate Error

Sample rate difference between reference and measured signal.

Note that a sample rate error is not available if the sample rate error estimation is switched off. See also Chapter 4.10, "Estimating and Compensating Signal Errors", on page 74.

FETCh:MACCuracy:SRERror:CURRent[:RESult]? on page 136

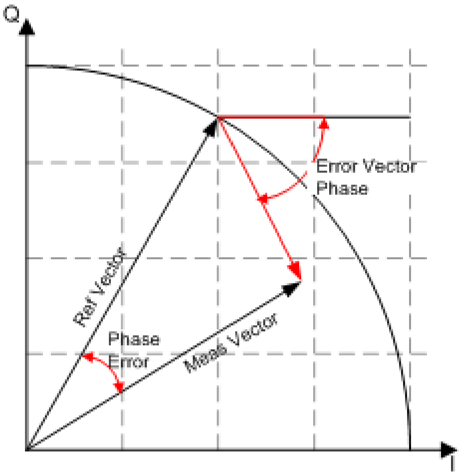
Magnitude Error

Difference in magnitude between the reference signal and the measured signal.

FETCh:MACCuracy:MERRor:CURRent[:RESult]? on page 134

Phase Error

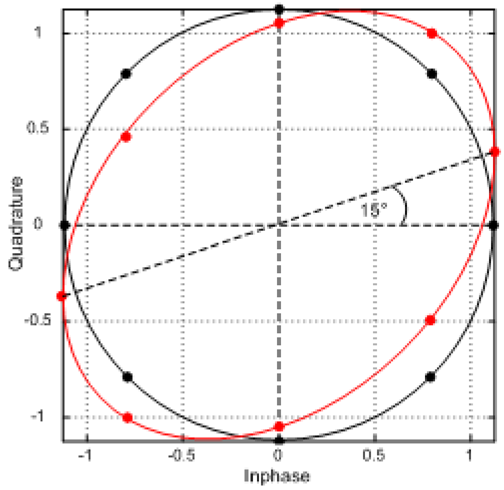
Phase difference between reference and measured signal.



FETCh:MACCuracy:PERRor:CURRent[:RESult]? on page 135

Quadrature Error

Phase deviation of the 90° phase difference between the real (I) and imaginary (Q) part of the signal.



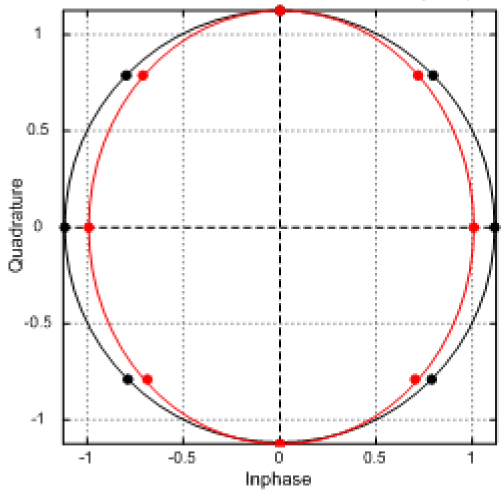
Within an ideal transmitter, the I and Q signal parts are mixed with an angle of 90° by the I/Q output mixer. Due to hardware imperfections, the signal delay of I and Q can be different and thus lead to an angle non-equal to 90°.

Note that quadrature rate error is not available if the I/Q Imbalance estimation is switched off. See also Chapter 4.10, "Estimating and Compensating Signal Errors", on page 74.

FETCh:MACCuracy:QERRor:CURRent[:RESult]? on page 135

Gain Imbalance

Gain difference between the real (I) and imaginary (Q) part of the signal.



This effect is typically generated by two separate amplifiers with a different gain in the I and Q path of the analog baseband signal generation.

Note that gain imbalance is not available if the I/Q Imbalance estimation is switched off. See also Chapter 4.10, "Estimating and Compensating Signal Errors", on page 74.

`FETCh:MACCuracy:GIMBalance:CURRent[:RESult]?` on page 133

I/Q Imbalance

Combination of Quadrature error and Gain imbalance.

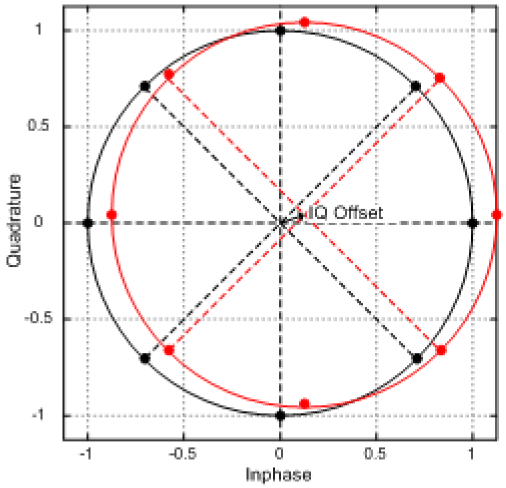
The I/Q imbalance parameter is a representation of the combination of Quadrature error and gain imbalance.

Note that I/Q imbalance is not available if the I/Q imbalance estimation is switched off. See also Chapter 4.10, "Estimating and Compensating Signal Errors", on page 74.

`FETCh:MACCuracy:IQIMbalance:CURRent[:RESult]?` on page 134

I/Q Offset

Shift of the measured signal compared to the ideal I/Q constellation in the I/Q plane.



Note that I/Q offset is not available if the I/Q Offset estimation is switched off. See also Chapter 4.10, "Estimating and Compensating Signal Errors", on page 74.

`FETCh:MACCuracy:IQOFfset:CURRent[:RESult]?` on page 134

Amplitude Droop

Amplitude droop is a measure of the change in magnitude of the signal over the frame (reference signal) being measured in dB.

Note that amplitude droop is not available if the amplitude droop estimation is switched off. See also Chapter 4.10, "Estimating and Compensating Signal Errors", on page 74.

Results to check power characteristics ← Numeric Result Summary

Power In

Signal power at the DUT input when reference signal is active. The signal generator level may change during direct DPD, but this result summary value will always refer to the reference signal – not the DPD signal.

`FETCh:POWer:INPut:CURRent[:RESult]?` on page 139

Power Out

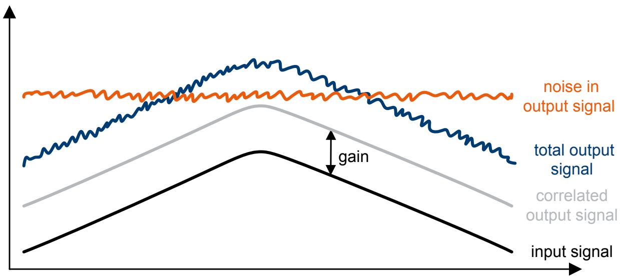
Signal power at the DUT output.
It is the RMS power of:

- The currently selected frame, if R&S FPS-K18 has successfully synchronized.
- The current capture buffer, if R&S FPS-K18 has not synchronized.

`FETCh:POWer:OUTPut:CURRent[:RESult]?` on page 140

Gain

Average gain calculated over all samples of the Gain Compression trace.



Note that gain is not necessarily equal to the ratio "Power Out" / "Power In". Gain only describes the ratio of the correlated signal in "Power Out" to "Power In".

Gain is always referenced to the reference signal power, i.e. when DPD changes the generator level, the gain is still referenced to the input power of the reference signal - not the DPD signal.

Example: If the output signal contains the same amount of noise as the correlated signal (e.g. signal is 0 dBm and noise power is also 0 dBm), "Power Out" will show the sum (3 dBm). However, assuming an input signal power of -10 dBm, gain will only show 10 dB, not 13 dB.

`FETCh:POWer:GAIN:CURRent[:RESult]?` on page 139

Crest Factor In

Crest factor of the signal at the DUT input. The crest factor is the ratio of the RMS and peak power.

`FETCh:POWer:CFACtor:IN:CURRent[:RESult]?` on page 138

Crest Factor Out

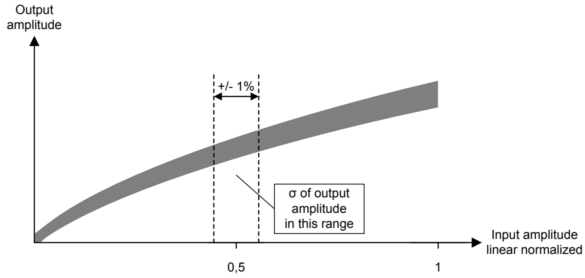
Crest factor of the signal at the DUT output. The crest factor is the ratio of the RMS and peak power.

`FETCh:POWer:CFACtor:OUT:CURRent[:RESult]?` on page 138

AM/AM Curve Width

Vertical spread of the samples in the AM/AM result display.

The AM/AM curve width shows the standard deviation of the output voltage or the output phase deviation within a +/- 1% range around the mean amplitude in volt.

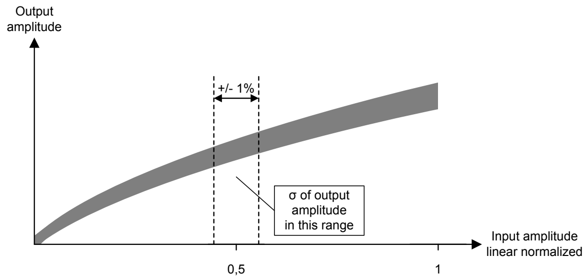


`FETCh:AMAM:CWIDth:CURRent[:RESult]?` on page 137

AM/PM Curve Width

Vertical spread of the samples in the AM/PM result display.

The AM/PM curve width shows the standard deviation of the output voltage or the output phase deviation within a +/- 1% range around the mean amplitude in volt.



`FETCh:AMPM:CWIDth:CURRent[:RESult]?` on page 138

Compression Point (1 dB / 2 dB / 3 dB)

Input power where the gain deviates by 1 dB, 2 dB or 3 dB from a reference gain (see "Configuring compression point calculation" on page 85).

In the graphical result, the compression points are indicated by horizontal red lines.

`FETCh:POWer:P1DB:CURRent[:RESult]?` on page 140

`FETCh:POWer:P2DB:CURRent[:RESult]?` on page 140

`FETCh:POWer:P3DB:CURRent[:RESult]?` on page 141

Output Compression Point (1 dB / 2 dB / 3 dB)

Output power where the gain deviates by 1 dB, 2 dB or 3 dB from a reference gain.

Uses identical operating points as "Compression Point (1 dB / 2 dB / 3 dB)", but is identified by output power at compression point rather than input power.

`FETCh:POWer:OUTPut:P1DB:CURRent[:RESult]?` on page 141

`FETCh:POWer:OUTPut:P2DB:CURRent[:RESult]?` on page 142

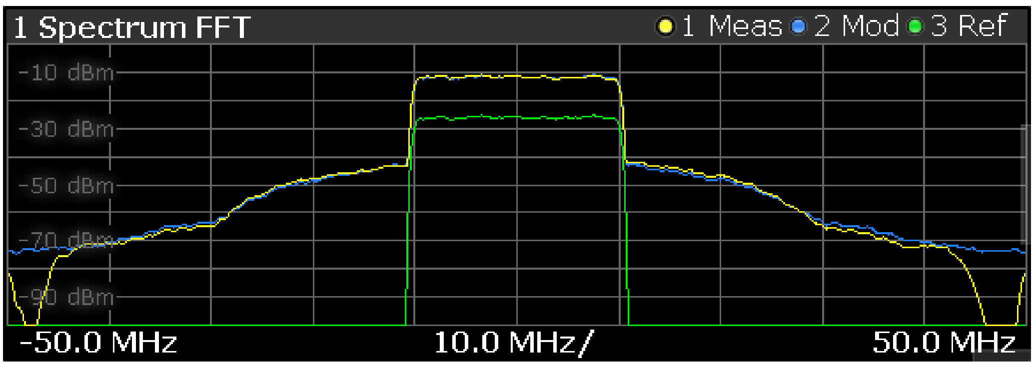
`FETCh:POWer:OUTPut:P3DB:CURRent[:RESult]?` on page 142

Spectrum FFT

The "Spectrum FFT" result display shows the frequency spectrum of the signal.

The spectrum FFT result shows the signal level in the spectrum around the center frequency. The unit is dBm.

You can display the spectrum of the measured signal and the reference signal. In the best case, the measured signal has the same shape as the reference signal.

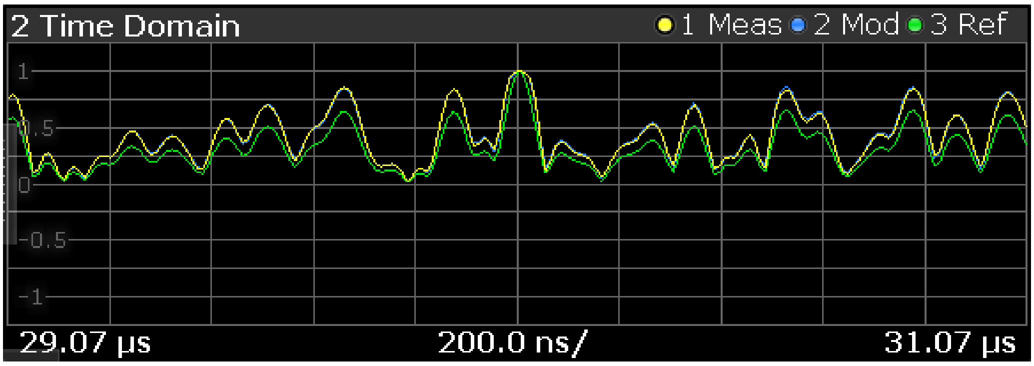


Remote command:
Selection (RF): `LAY:ADD? '1',LEFT,RFS`
Result query: `TRACe<n>[:DATA]?` on page 129

Time Domain
The "Time Domain" result display shows the signal characteristics over time.

It is similar to the "Power vs Time" and "Magnitude Capture" result displays in that it shows the signal characteristics over time. However, it deliberately shows only a very short period of the signal. You can thus use it to compare various aspects of the signal, especially the timing of the displayed signals, in a single result display.

- Measured signal
  Trace 1 shows the characteristics of the measured signal over time. The data should be the same as the results shown in the Magnitude Capture RF result display.
  In the best case, the measured signal is the same as the reference signal.
- Modeled signal
  Trace 2 shows the characteristics of the modeled signal. When system modeling has been turned off, this trace is not displayed.
  If the model matches the behavior of the DUT, the characteristics of the signal are the same as those of the measured signal (minus the noise).
- Reference signal
  Trace 3 shows the characteristics of the reference signal. The reference signal present at the DUT input represents the ideal signal.



Remote command:
Selection: `LAY:ADD? '1',LEFT,TDOM`
Result query: `TRACe<n>[:DATA]?` on page 129

Scale of the x-axis (display settings for the time domain) ← Time Domain
The scale of the x-axis depends on your configuration in the "Display Settings" dialog box.

The logic is as follows:
- When you select automatic scaling (→ "Position: Auto") and synchronization has failed, the application searches for the peak level in the capture buffer and shows the signal around the peak for the "Duration" that has been defined.
- When you select automatic scaling (→ "Position: Auto") and synchronization is OK, the application searches for the peak level in the synchronized area of the capture

buffer and shows the signal around the peak for the "Duration" that has been defined.

- When you select manual scaling (→ "Position: Manual") and synchronization has failed, the x-axis starts at an "Offset" relative to the first sample in the capture buffer. The end of the x-axis depends on the "Duration" you have defined.
- When you select manual scaling (→ "Position: Manual") and synchronization is OK, the x-axis starts at an "Offset" relative to the first sample in the synchronized area of the capture buffer. The end of the x-axis depends on the "Duration" you have defined.

Note: The "Display Settings" for the time domain are only available after you have selected the "Specifics for: Time Domain" item from the corresponding dropdown menu at the bottom of the dialog box.

Scale of the y-axis (display settings for the time domain) ← **Time Domain**

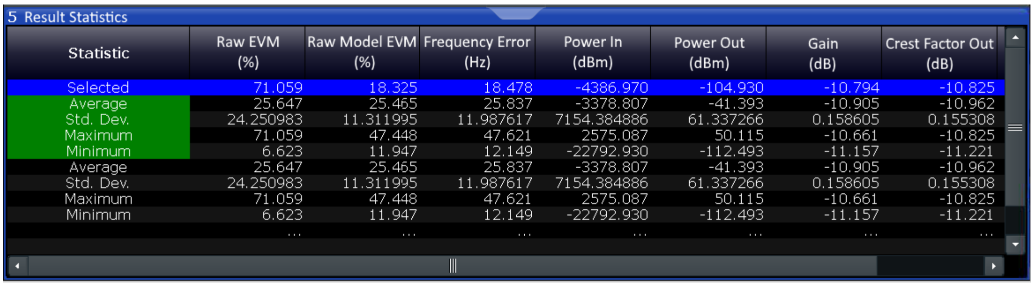
The scale of the y-axis also depends on your configuration.

The signal characteristics displayed in the time domain result display all have a different unit. Therefore, the application provides a feature that normalizes all results to 1 (see "Configuring the time domain result display" on page 102). Normalization makes it easier to compare the timing between the traces. By default, normalization is on. Note that you can normalize each Time Domain window individually.

Unnormalized results are displayed in their respective unit.

Statistics Table

The results for the statistics table are available only after the statistics mode has been activated using `[SENSe:]SWEep:STATistics[:STATe]` on page 245. If statistics mode is switched off, the statistics table stays empty.



5 Result Statistics

Statistic	Raw EVM (%)	Raw Model EVM (%)	Frequency Error (Hz)	Power In (dBm)	Power Out (dBm)	Gain (dB)	Crest Factor Out (dB)
Selected	71.059	18.325	18.478	-4386.970	-104.930	-10.794	-10.825
Average	25.647	25.465	25.837	-3378.807	-41.393	-10.905	-10.962
Std. Dev.	24.250983	11.311995	11.987617	7154.384886	61.337266	0.158605	0.155308
Maximum	71.059	47.448	47.621	2575.087	50.115	-10.661	-10.825
Minimum	6.623	11.947	12.149	-22792.930	-112.493	-11.157	-11.221
Average	25.647	25.465	25.837	-3378.807	-41.393	-10.905	-10.962
Std. Dev.	24.250983	11.311995	11.987617	7154.384886	61.337266	0.158605	0.155308
Maximum	71.059	47.448	47.621	2575.087	50.115	-10.661	-10.825
Minimum	6.623	11.947	12.149	-22792.930	-112.493	-11.157	-11.221
	...	...	...	...	...	...	...

Each value in the statistics table has different rows describing a single frame: Average, Std. Dev, Maximum and Minimum. This is similar to the Numeric Result Summary.

The different color codes represent different result values:

- **Blue**
  Result of the current result range. The selected values are updated when the user sweeps through the result range selection.
- **Green**
  In I/Q averaging mode, the values in the green area are identical to the ones in the black background area.

In trace statistics mode, the green area refers to all frames of the current capture buffer, whereas the black area refers to all measured frames (including previous capture buffers). Statistics is always done over sweep “Count” frames and then is being reset, unless the moving average switch is activated. In this case, infinite statistics is executed.

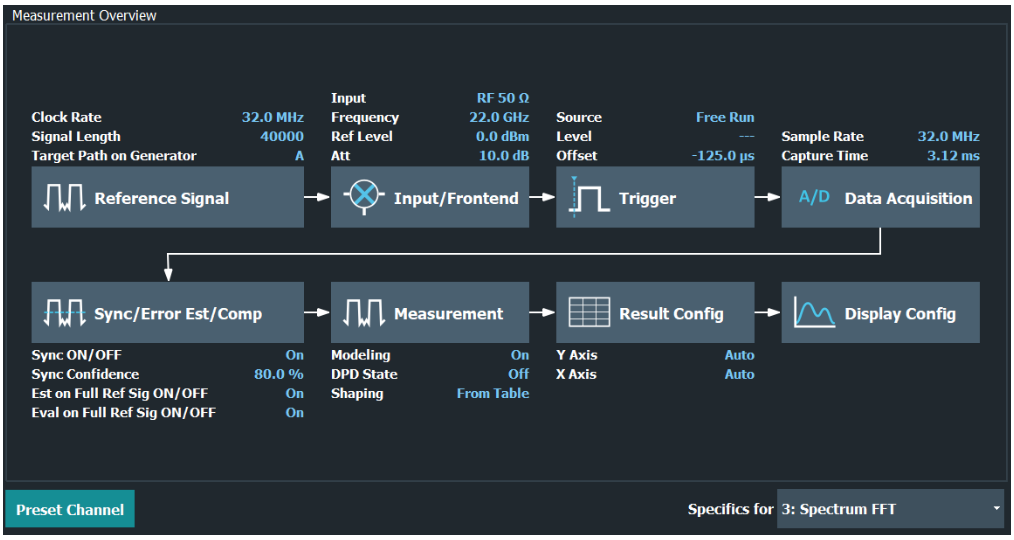
- **Black / No selection**
  Statistical results that can also be based on result ranges that were captured in previous measurement sweeps.

Remote command:
Adding statistics table: `LAY:ADD? '1',LEFT,STAB`
Querying results: Chapter 6.5.3.4, "Retrieving Results of the Statistics Table", on page 154
Configuring statistics table: Chapter 6.7.4, "Configuring the Statistics Table", on page 299
Navigating through results ranges found in a capture: `CONFigure:RESult:RANGe[:SELected]` on page 245

4 Configuration

4.1 Configuration Overview

Throughout the measurement channel configuration, an overview of the most important currently defined settings is provided in the "Overview". The "Overview" is displayed when you select the "Overview" icon, which is available at the bottom of all softkey menus.



In addition to the main measurement settings, the "Overview" provides quick access to the main settings dialog boxes. The individual configuration steps are displayed in the order of the data flow. Thus, you can easily configure an entire measurement channel from input over processing to output and analysis by stepping through the dialog boxes as indicated in the "Overview".

In particular, the "Overview" provides quick access to the following configuration dialog boxes (listed in the recommended order of processing):

1. Reference Signal
   See Chapter 4.3, "Designing a Reference Signal", on page 35.
2. Input and output
   See Chapter 4.4, "Configuring Inputs and Outputs", on page 43.
3. Trigger
   See Chapter 4.5, "Triggering Measurements", on page 65.
4. Data Acquisition
   See Chapter 4.6, "Configuring the Data Capture", on page 65.
5. Synchronization, error estimation and compensation
   See Chapter 4.8, "Synchronizing Measurement Data", on page 69.
   See Chapter 4.10, "Estimating and Compensating Signal Errors", on page 74.
6. Measurement
   Modeling: see Chapter 4.12, "Applying System Models", on page 76.
   DPD: see Chapter 4.13, "Applying Digital Predistortion", on page 79.
7. Result configuration
   See Chapter 5, "Analysis", on page 92.
8. Display configuration
   See Chapter 3, "Measurements and Result Displays", on page 11.

To configure settings

► Select any button in the "Overview" to open the corresponding dialog box.
  Select a setting in the channel bar (at the top of the measurement channel tab) to change a specific setting.

Preset Channel
Select the "Preset Channel" button in the lower left-hand corner of the "Overview" to restore all measurement settings **in the current channel** to their default values.

Do not confuse the "Preset Channel" button with the [Preset] *key*, which restores the entire instrument to its default values and thus closes **all channels** on the R&S FPS (except for the default channel)!

Remote command:
`SYSTem:PRESet:CHANnel[:EXEC]` on page 117

Specific Settings for
The channel may contain several windows for different results. Thus, the settings indicated in the "Overview" and configured in the dialog boxes vary depending on the selected window.

Select an active window from the "Specific Settings for" selection list that is displayed in the "Overview" and in all window-specific configuration dialog boxes.

The "Overview" and dialog boxes are updated to indicate the settings for the selected window.

4.2 Performing Measurements

Access: [SWEEP]

The following features control the measurement. They are available in the "Sweep" menu.

The remote commands required to control the measurement are described in Chapter 6.5.1, "Performing Measurements", on page 125.

Continuous Sweep / Run Cont
After triggering, starts the measurement and repeats it continuously until stopped.

While the measurement is running, the "Continuous Sweep" softkey and the [RUN CONT] key are highlighted. The running measurement can be aborted by selecting the highlighted softkey or key again. The results are not deleted until a new measurement is started.

Note: Sequencer. If the Sequencer is active, the "Continuous Sweep" softkey only controls the sweep mode for the currently selected channel. However, the sweep mode only takes effect the next time the Sequencer activates that channel, and only for a channel-defined sequence. In this case, a channel in continuous sweep mode is swept repeatedly.
Furthermore, the [RUN CONT] key controls the Sequencer, not individual sweeps. [RUN CONT] starts the Sequencer in continuous mode.

Remote command:
`INITiate<n>:CONTinuous` on page 126

Single Sweep / Run Single
After triggering, starts the number of sweeps set in "Sweep Count". The measurement stops after the defined number of sweeps has been performed.

While the measurement is running, the "Single Sweep" softkey and the [RUN SINGLE] key are highlighted. The running measurement can be aborted by selecting the highlighted softkey or key again.

Remote command:
`INITiate<n>[:IMMediate]` on page 126

Continue Single Sweep
While the measurement is running, the "Continue Single Sweep" softkey and the [RUN SINGLE] key are highlighted. The running measurement can be aborted by selecting the highlighted softkey or key again.

Remote command:
`INITiate<n>:CONMeas` on page 125

4.3 Designing a Reference Signal

Access (source: generator): "Overview" > "Reference Signal" > "Current Generator Waveform"

Access (source: waveform file): "Overview" > "Reference Signal" > "Custom Waveform File"

Access (source: Amplifier application): "Overview" > "Reference Signal" > "Generate Own Signal"

Many of the results available in the application require a reference signal that describes the characteristics of the signal you feed into the amplifier.

The reference signal describes the characteristics of the signal that you feed into the amplifier and whose amplified version is measured by the application. You can define any signal you want as a reference signal.

The application provides several methods to design a reference signal:

- Designing the signal on a generator
  (Having a Rohde & Schwarz generator is mandatory for this method.)
- Designing the signal in a waveform file
- Designing the signal in the amplifier application
  (Having a Rohde & Schwarz generator is mandatory for this method.)

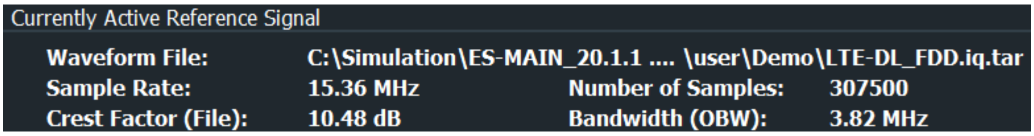
For a list of supported signal generators, refer to the datasheet of the amplifier application.

The remote commands required to configure the reference signal are described in Chapter 6.6.1, "Designing a Reference Signal", on page 209.

Reference signal information

Each tab of the "Reference Signal" dialog box contains some basic information about the reference signal currently in use.

The information is only displayed when a reference signal has been successfully loaded. When you load a different waveform, the reference signal information is updated accordingly.



- Waveform file
  Name and path of the waveform file currently in use.
- Sample rate
  The sample rate in the header of the currently used reference signal waveform file in Hz.
- Number of samples
  Length of the currently used reference signal waveform file in samples.
- Crest Factor (File)
  Crest factor of the whole file currently in use. The crest factor of waveform files is read from their header. The crest factor of iq.tar files is calculated.
- Bandwidth (OBW)
  The occupied bandwidth of the reference signal currently in use. A calculated bandwidth that contains 99% of signal power is displayed.

Remote command:
File path: `CONFigure:REFSignal:SINFo:FPATh?` on page 216
Sample rate: `CONFigure:REFSignal:SINFo:SRATe?` on page 217
Sample length: `CONFigure:REFSignal:SINFo:SLENgth?` on page 217
Crest Factor: `CONFigure:REFSignal:SINFo:CFACtor?` on page 217
OBW: `CONFigure:REFSignal:SINFo:OBW?` on page 218

Using multi-segment waveform files

Modern chip technologies implement several communication standards within one chip and thus increase the requirements in spatial design and test systems. To fulfill the requirements in the test systems, and to enable a rapid change between different waveforms containing different test signals, the R&S SMW provides the functionality to generate multi-segment waveform files. Multi-segment waveform files are files that contain several different waveforms.

(For more information about creating and using multi-segment waveform files (including examples) refer to the documentation of the R&S SMW.)

When you are testing amplifiers with the amplifier measurement application, you can use a multi-segment waveform file to create the reference signal. If you use one of these files, you have to select the segment that you want to use as a reference signal in the corresponding input field.

Note that the content of the segment you are using for the reference signal must match the content of the segment used by the ARB of the signal generator. You can select the segment for the used by the generator in the generator setup.

Remote command:
`CONFigure:REFSignal:SEGMent` on page 216

Transferring the reference signal
Both the signal generator and analyzer used in the test setup need to know the characteristics of the reference signal.

- The signal generator needs that information to generate the signal.
- The analyzer needs that information for the evaluation of the results.

This is why you have to transfer the signal information to both instruments. The transmission is done through a LAN connection that you have to establish when setting up the measurement. For more information on that see Chapter 4.4.5, "Controlling a Signal Generator", on page 50.

- When you design the reference signal on the signal generator, transfer the signal information from the generator to the analyzer with the ➙"Read and Load Current Signal from R&S SMW" button.
  You can either design a reference signal with one of the available firmware options (for example an LTE signal with the R&S SMW-K55) or design a signal in a custom waveform file. Note that the R&S FPS-K18 does not support all firmware options of the signal generator.
- When you load the reference signal from a waveform file or design the signal within the R&S FPS-K18, transfer the signal information from the analyzer to the generator. Depending on the signal source, you can do this either with the "Load and Export Selected Waveform File to Generator" or the "Generate and Load Signal and Export it to Generator" buttons.

When you send the signal information to the generator, the application automatically configures the generator accordingly.

Transmission state

The LED displayed with the transmission button shows the state of the reference signal transmission.

The LED is either gray, green or red:
- Grey LED
  Transmission state unknown (for example when you have not yet started the transmission).
- Green LED
  Transmission has been successful.
- Red LED
  Transmission has not been successful.
  Make sure that the generator control state is on. Also check if the generator IP address / computer name are correct and if the connection has been established.

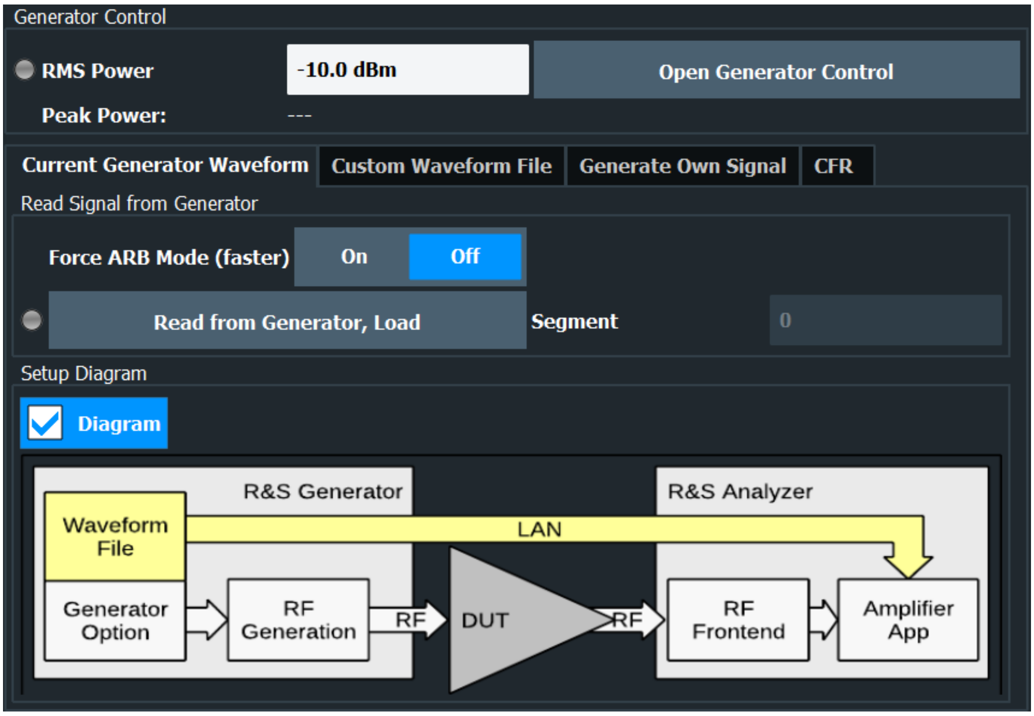
Designing a reference signal on a signal generator
One way to design a reference signal is to design the signal on the signal generator itself.

You can design any signal you like, as long as it is storable as an arbitrary waveform (ARB) file. When you are done, you have to transfer the signal information from the signal generator to the signal analyzer with the "Read from Generator, Load" button.

The "Force ARB Mode" switch forces the signal generator to use its ARB mode (arbitrary waveform) rather than its real-time mode, whenever possible. As a result, switching between DPD on and off state is significantly faster. When the "Force ARB Mode" function is used, the peak power of the generator is read out and used within the process but as a result of this function the RMS power of the generator is modified.

The parameters of the currently active reference signal are described in "Reference signal information" on page 36.

The "Open Generator Control" button provides functionality to change the generator settings as described in Chapter 4.4.5, "Controlling a Signal Generator", on page 50.



Most of the options available for the connected generator are supported by the automatic signal import functionality of the R&S FPS-K18. If the signal import was not successful (indicated by a red LED), you have to transfer the reference signal in another way (for example with a memory stick).

For a comprehensive description of all features available on the signal generator and information on how to generate signals, refer to the documentation of the signal generator.

Remote command:
See signal generator documentation.
`CONFigure:REFSignal:CGW:AMODe[:STATe]` on page 210
`CONFigure:REFSignal:CGW:READ` on page 210
`CONFigure:REFSignal:CGW:LEDState?` on page 210

Designing a reference signal in a waveform file
One way to design a reference signal is to define its characteristics in a waveform file (`*.wv` or `*.iq.tar`).

You can create a waveform file, for example:
- With the R&S®WinIQSIM2 software package
- By exporting a signal designed on the signal generator

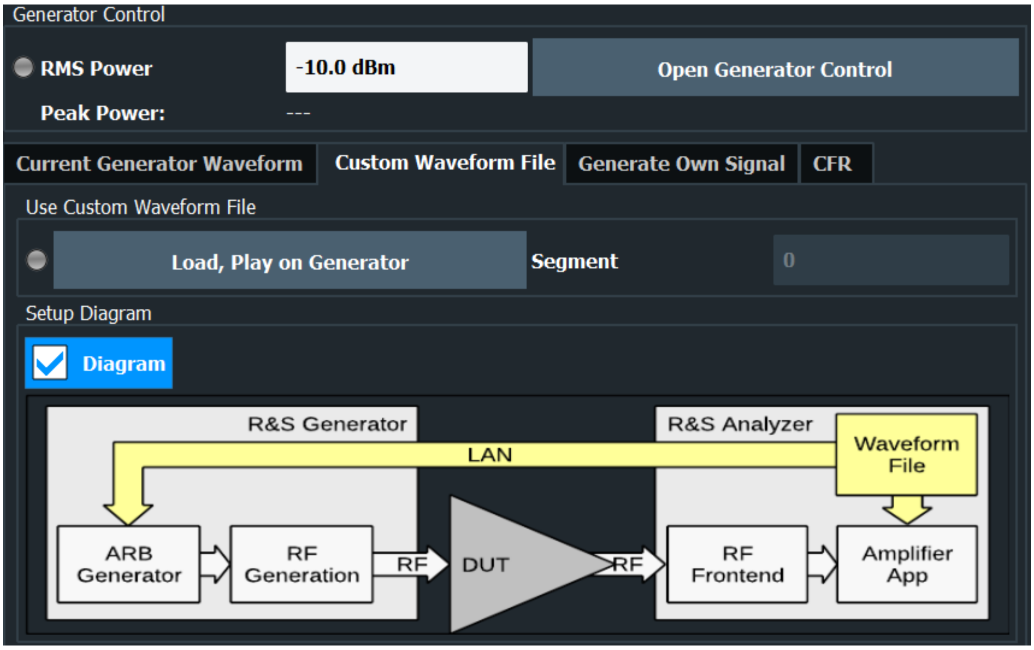
Basically, this file contains the characteristics of the reference signal. The generator then generates the reference signal based on the information in the file.

There are two ways to generate the reference signal through a custom waveform file.
- The generator is connected to the R&S FPS in a LAN, and can be recognized by the R&S FPS-K18 (Rohde & Schwarz generators only, for example the R&S SMW) In that case, you can simply transfer the reference signal information to the generator with the features integrated into the R&S FPS-K18. The generator then generates the corresponding signal with the appropriate signal level, and the R&S FPS-K18 is able to compare the measured signal to the ideal reference signal.
- The generator is not connected to the R&S FPS In that case, you have to load the reference signal information onto the generator manually and turn off the "Export to Generator" function. Because no exchange of information is possible between generator and analyzer, it is required to specify the input level of the signal in the "DUT Peak Input Power" input field.

The parameters of the currently active reference signal are described in "Reference signal information" on page 36.

The "Open Generator Control" button provides functionality to change the generator settings as described in Chapter 4.4.5, "Controlling a Signal Generator", on page 50.

For a comprehensive description of all features available on the signal generator and information on how to generate and export signals to a file, refer to the documentation of the signal generator.



To transfer a waveform file from the analyzer to the generator and process it with the ARB generator of the R&S SMW, for example, proceed as follows:

► In the "Custom Waveform" tab, select a file via the "Load, Play on Generator" button.

► Transfer the file to the generator with the "Select" button.

If a waveform is only used as a reference without transferring it to the signal generator, make sure that the generator control state "Off" is selected in the generator setup dialog.

Remote command:
Select file: `CONFigure:REFSignal:CWF:FPATh` on page 212
Transfer file: `CONFigure:REFSignal:CWF:WRITe` on page 212
Transmission state: `CONFigure:REFSignal:CWF:LEDState?` on page 212
Export file: `CONFigure:REFSignal:CWF:ETGenerator[:STATe]` on page 211
DUT input power: `CONFigure:REFSignal:CWF:DPIPower` on page 211

Designing a reference signal within the R&S FPS-K18

One way to design a reference signal is to design the signal within the R&S FPS-K18.

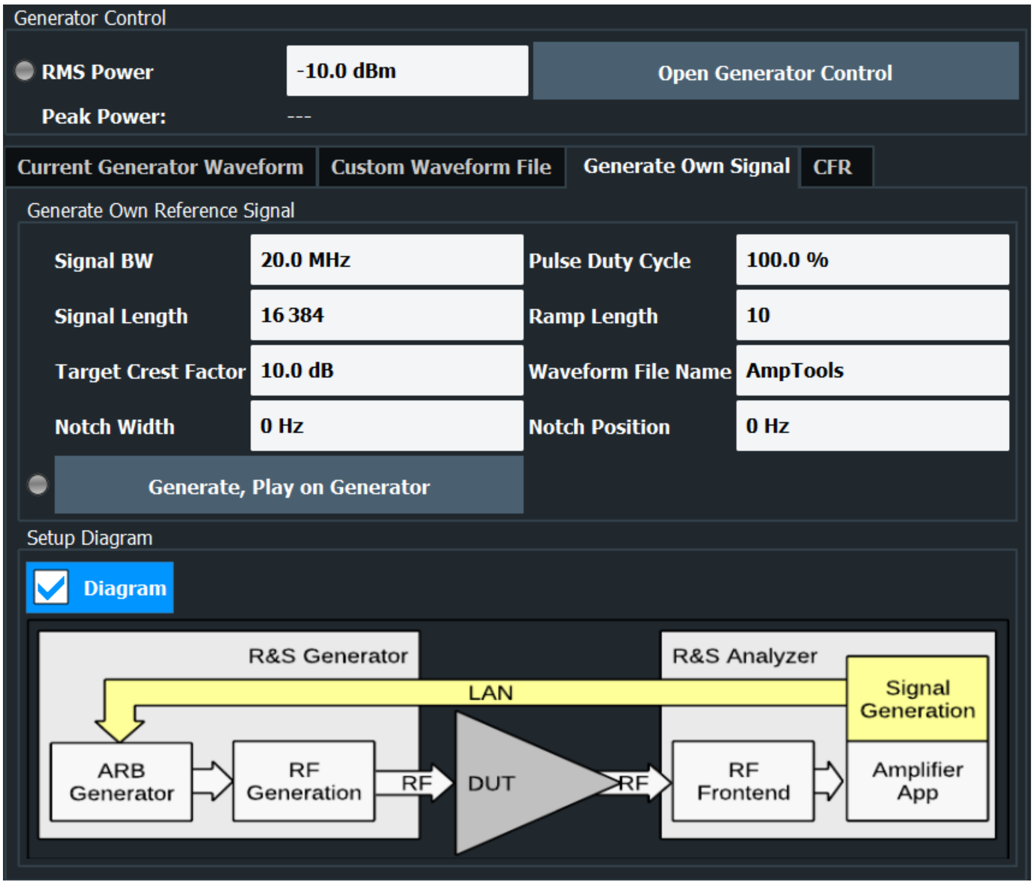
The application provides functionality to design a basic reference signal and saves the signal characteristics in a waveform file which you have to transfer to the signal generator with the "Generate and Load Signal and Export it to Generator" button.

When the data has been transferred, the signal generator (for example the R&S SMW) generates the corresponding signal.

The generated signal is a multi-carrier CW signal, whose basic properties, like crest factor and bandwidth, you can specify as required.

The parameters of the currently active reference signal are described in "Reference signal information" on page 36.

The "Open Generator Control" button provides functionality to change the generator settings as described in Chapter 4.4.5, "Controlling a Signal Generator", on page 50.



To generate a reference signal within the application, proceed as follows:

► In the "Generate Own Signal" tab, design the reference signal as required.

The application stores the current signal properties as an ARB signal in a waveform file.

► Upload the data to the generator with the "Generate, Play on Generator" button.

You can define the following signal characteristics.
- "Signal Bandwidth" on page 41
- "Pulse Duty Cycle" on page 42
- "Signal Length" on page 42
- "Ramp Length" on page 42
- "Crest Factor" on page 42
- "Waveform File Name" on page 42
- "Notch Width" on page 42
- "Notch Position" on page 43

Remote command:
`CONFigure:REFSignal:GOS:WRITe` on page 216
`CONFigure:REFSignal:GOS:LEDState?` on page 214

Signal Bandwidth ← Designing a reference signal within the R&S FPS-K18
Defines the bandwidth of the reference signal.

The bandwidth should not be larger than maximum I/Q bandwidth supported by your signal analyzer (which depends on the analyzer configuration).

Remote command:
`CONFigure:REFSignal:GOS:BWIDth` on page 213

Pulse Duty Cycle ← Designing a reference signal within the R&S FPS-K18
Defines the duty cycle of a pulsed reference signal.

The duty cycle of a pulse is the ratio of the pulse duration and the actual length of the pulse. A duty cycle of 100 % corresponds to a continuous signal.

Example:

The pulse duration is 2 µs. The actual length of the pulse is 1 µs. The duty cycle is 1 µs : 2 µs = 0.5 or 50 %.

Remote command:
`CONFigure:REFSignal:GOS:DCYCle` on page 213

Signal Length ← Designing a reference signal within the R&S FPS-K18
Defines the number of samples that the reference signal consists of.

A number that is a power of 2 speeds up the internal signal processing. Thus, such a number should be specified if no other requirements limit the choice of the sample count.

For more information, see "Pulse Duty Cycle" on page 42.

Remote command:
`CONFigure:REFSignal:GOS:SLENgth` on page 215

Ramp Length ← Designing a reference signal within the R&S FPS-K18
Defines the number of samples used to ramp up the pulse to its full power and vice versa.

Remote command:
`CONFigure:REFSignal:GOS:RLENgth` on page 215

Crest Factor ← Designing a reference signal within the R&S FPS-K18
Defines the crest factor of the reference signal.

The crest factor shows the RMS power in relation to the peak power.

Remote command:
`CONFigure:REFSignal:GOS:CRESt` on page 213

Waveform File Name ← Designing a reference signal within the R&S FPS-K18
Defines the name of the waveform file that the reference ARB signal configuration is stored in.

Remote command:
`CONFigure:REFSignal:GOS:WNAMe` on page 215

Notch Width ← Designing a reference signal within the R&S FPS-K18
Defines the width of a notch that you can add to the reference signal.

Within the notch, all carriers of the reference signal have zero amplitude. You can use the noise notch to, for example, determine the noise power ratio (NPR) before and after the DPD.

Remote command:
`CONFigure:REFSignal:GOS:NWIDth` on page 214

Notch Position ← Designing a reference signal within the R&S FPS-K18
Defines an offset for the noise notch relative to the center frequency.

The offset moves the notch to a position outside the center of the signal. You can use the offset to, for example, generate a one-sided noise signal or to examine asymmetric distortion effects.

Remote command:
`CONFigure:REFSignal:GOS:NPOSition` on page 214

4.4 Configuring Inputs and Outputs

4.4.1 Selecting and Configuring the Input Source

Access: "Overview" > "Input / Frontend" > "Input Source"

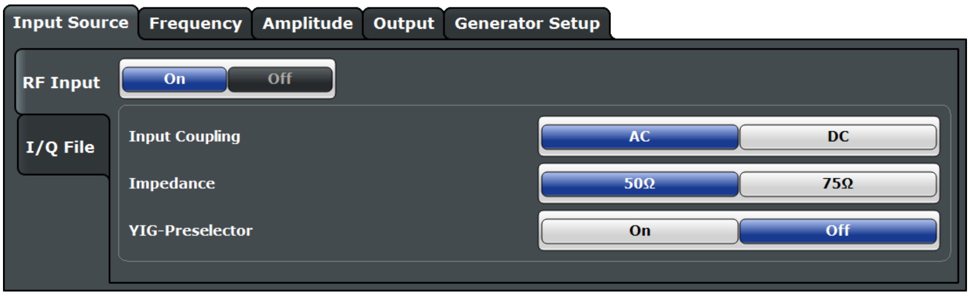
The R&S FPS-K18 supports the RF input.

4.4.1.1 Configuring the RF Input

Access: "Overview" > "Input / Frontend" > "Input Source" > "RF Input"

The RF input captures the RF signal that you are measuring. It is always on.

The RF input source settings are similar to those available in the spectrum application. For a comprehensive description of these settings, refer to the R&S FPS user manual.



The remote commands required to configure the RF input are described in Chapter 6.6.2, "Selecting and Configuring the Input Source", on page 223.

Input Coupling

The RF input of the R&S FPS can be coupled by alternating current (AC) or direct current (DC).

AC coupling blocks any DC voltage from the input signal. This is the default setting to prevent damage to the instrument. Very low frequencies in the input signal may be distorted.

However, some specifications require DC coupling. In this case, you must protect the instrument from damaging DC input voltages manually. For details, refer to the data sheet.

Remote command:
`INPut<ip>:COUPling` on page 223

Impedance

For some measurements, the reference impedance for the measured levels of the R&S FPS can be set to 50 Ω or 75 Ω.

Select 75 Ω if the 50 Ω input impedance is transformed to a higher impedance using a 75 Ω adapter of the RAZ type. (That corresponds to 25Ω in series to the input impedance of the instrument.) The correction value in this case is 1.76 dB = 10 log (75Ω/50Ω).

Remote command:
`INPut<ip>:IMPedance` on page 225

YIG-Preselector

Enables or disables the YIG-preselector, if available on the R&S FPS.

An internal YIG-preselector at the input of the R&S FPS ensures that image frequencies are rejected. However, this is only possible for a restricted bandwidth. To use the maximum bandwidth for signal analysis you can disable the YIG-preselector at the input of the R&S FPS, which can lead to image-frequency display.

Note that the YIG-preselector is active only on frequencies greater than 8 GHz. Therefore, switching the YIG-preselector on or off has no effect if the frequency is below that value.

Remote command:
`INPut<ip>:FILTer:YIG[:STATe]` on page 224

4.4.1.2 External Mixer

Access: "Overview" > "Input / Frontend" > "Input Source" > "External Mixer"

Controlling external generators is available with the optional external generator control. The functionality is the same as in the spectrum application.

For more information about using external generators, refer to the R&S FPS user manual.

4.4.1.3 I/Q File

Access: "Overview" > "Input / Frontend" > "Input Source" > "I/Q File"

Available for I/Q based measurements.

As an alternative to capturing the measurement (I/Q) data live, you can also load previously recorded I/Q data stored in an `iq.tar` file. The file is then used as the input source for the application. Files containing multi-channel measurement data are supported.

However, only the RF capture buffer is filled with I/Q data from the file. Envelope / supply power measurements based on data from the analog baseband interface (R&S FPS-B71) are not supported in I/Q file mode.

For details on the "I/Q File" input, see the user manual of the I/Q analyzer.

I/Q Input File State

Enables input from the selected I/Q input file.

If enabled, the application performs measurements on the data from this file. Thus, most measurement settings related to data acquisition (attenuation, center frequency, measurement bandwidth, sample rate) cannot be changed. The measurement time can only be decreased, to perform measurements on an extract of the available data only.

Note: Even when the file input is disabled, the input file remains selected and can be enabled again quickly by changing the state.

Remote command:
`INPut<ip>:SELect` on page 225

Select I/Q data file

Opens a file selection dialog box to select an input file that contains I/Q data.

The I/Q data must have a specific format (`.iq.tar`) as described in R&S FPS I/Q Analyzer and I/Q Input User Manual.

The default storage location for I/Q data files is `C:\R_S\INSTR\USER`.

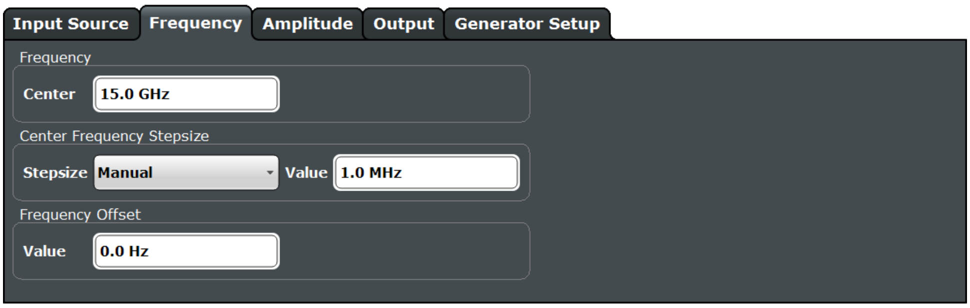
Remote command:
INPut<ip>:FILE:PATH on page 224

4.4.2 Configuring the Frequency

Access: "Overview" > "Input / Output" > "Frequency"

The "Frequency" tab of the "Input / Output" dialog box contains settings to configure frequency characteristics.

The frequency settings are similar to those available in the spectrum application. For a comprehensive description of these settings, refer to the R&S FPS user manual.



The remote commands required to configure the frequency are described in Chapter 6.6.3, "Configuring the Frequency", on page 225.

Center Frequency
Defines the frequency of the measured signal.

The possible value range depends on the R&S FPS model you have. See the data sheet for more information about the supported frequency range.

Remote command:
[SENSe:]FREQuency:CENTer on page 225

Center Frequency Stepsize
Defines the step size by which the center frequency is increased or decreased when the arrow keys are pressed.

When you use the rotary knob the center frequency changes in steps of only 1/10 of the "Center Frequency Stepsize".

"= Center"	Sets the step size to the value of the center frequency and removes the coupling of the step size to span or resolution bandwidth. The used value is indicated in the "Value" field.
"Manual"	Defines a fixed step size for the center frequency. Enter the step size in the "Value" field.

Remote command:
[SENSe:]FREQuency:CENTer:STEP on page 226

Frequency Offset
Shifts the displayed frequency range along the x-axis by the defined offset.

This parameter has no effect on the instrument's hardware, or on the captured data or on data processing. It is simply a manipulation of the final results in which absolute frequency values are displayed. Thus, the x-axis of a spectrum display is shifted by a constant offset if it shows absolute frequencies. However, if it shows frequencies relative to the signal's center frequency, it is not shifted.

A frequency offset can be used to correct the display of a signal that is slightly distorted by the measurement setup, for example.

The allowed values range from -1 THz to 1 THz. The default setting is 0 Hz.

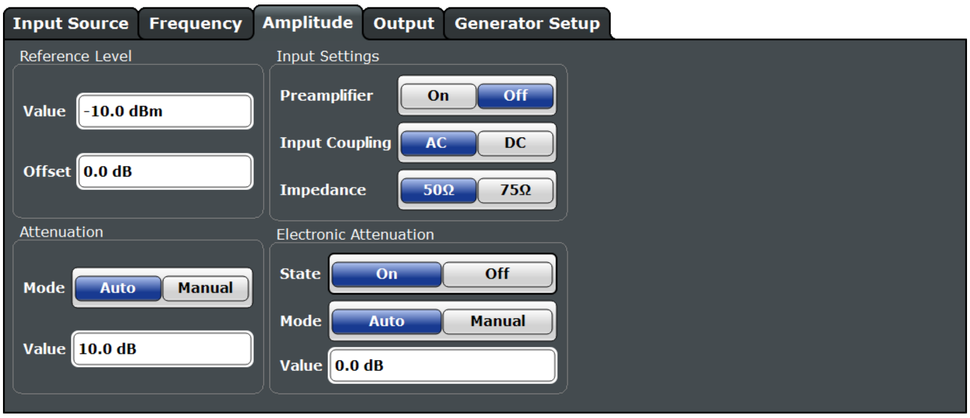
Remote command:
`[SENSe:]FREQuency:OFFSet` on page 226

4.4.3 Defining Level Characteristics

Access: "Overview" > "Input / Output" > "Amplitude"

The "Amplitude" tab of the "Input / Output" dialog box contains settings to configure the signal level characteristics.

The level settings are the same as those available in the spectrum application. For a comprehensive description of these settings, refer to the R&S FPS user manual.



The remote commands required to configure the amplitude are described in Chapter 6.6.4, "Defining Level Characteristics", on page 226.

Functions available in the "Amplitude" dialog box described elsewhere:

- " Input Coupling " on page 44
- " Impedance " on page 44

Reference Level

Defines the expected maximum reference level. Signal levels above this value may not be measured correctly. This is indicated by an "IF Overload" status display.

The reference level can also be used to scale power diagrams; the reference level is then used as the maximum on the y-axis.

Since the hardware of the R&S FPS is adapted according to this value, it is recommended that you set the reference level close above the expected maximum signal level. Thus you ensure an optimum measurement (no compression, good signal-to-noise ratio).

Remote command:
`DISPlay[:WINDow<n>]:TRACe<t>:Y[:SCALe]:RLEVel` on page 227

Shifting the Display (Offset) ← Reference Level

Defines an arithmetic level offset. This offset is added to the measured level. In some result displays, the scaling of the y-axis is changed accordingly.

Define an offset if the signal is attenuated or amplified before it is fed into the R&S FPS so the application shows correct power results. All displayed power level results are shifted by this value.

The reference level offset takes level offsets into account that occur after the signal has passed through the DUT (usually an amplifier). For level offsets occurring before the DUT, you can define a level offset on the signal generator from within the R&S FPS-K18 user interface.

The setting range is ±200 dB in 0.01 dB steps.

Note, however, that the *internal* reference level (used to adjust the hardware settings to the expected signal) ignores any "Reference Level Offset" . Thus, it is important to keep in mind the actual power level the R&S FPS must handle. Do not rely on the displayed reference level (internal reference level = displayed reference level - offset).

Remote command:
`DISPlay[:WINDow<n>]:TRACe<t>:Y[:SCALe]:RLEVel:OFFSet` on page 227

Preamplifier (option B22/B24)

Switches the preamplifier on and off. If activated, the input signal is amplified by 20 dB.

If option R&S FPS-B22 is installed, the preamplifier is only active below 7 GHz.

If option R&S FPS-B24 is installed, the preamplifier is active for all frequencies.

Remote command:
`INPut<ip>:GAIN:STATe` on page 229

Input Coupling

The RF input of the R&S FPS can be coupled by alternating current (AC) or direct current (DC).

AC coupling blocks any DC voltage from the input signal. This is the default setting to prevent damage to the instrument. Very low frequencies in the input signal may be distorted.

However, some specifications require DC coupling. In this case, you must protect the instrument from damaging DC input voltages manually. For details, refer to the data sheet.

Remote command:
`INPut<ip>:COUPling` on page 223

Impedance
For some measurements, the reference impedance for the measured levels of the R&S FPS can be set to 50 Ω or 75 Ω.

Select 75 Ω if the 50 Ω input impedance is transformed to a higher impedance using a 75 Ω adapter of the RAZ type. (That corresponds to 25Ω in series to the input impedance of the instrument.) The correction value in this case is 1.76 dB = 10 log (75Ω/50Ω).

Remote command:
`INPut<ip>:IMPedance` on page 225

Attenuation Mode / Value
The RF attenuation can be set automatically as a function of the selected reference level (Auto mode). This ensures that no overload occurs at the RF Input connector for the current reference level. It is the default setting.

By default and when no (optional) electronic attenuation is available, mechanical attenuation is applied.

In "Manual" mode, you can set the RF attenuation in 1 dB steps (down to 0 dB). Other entries are rounded to the next integer value. The range is specified in the data sheet. If the defined reference level cannot be set for the defined RF attenuation, the reference level is adjusted accordingly and the warning "limit reached" is displayed.

NOTICE! Risk of hardware damage due to high power levels. When decreasing the attenuation manually, ensure that the power level does not exceed the maximum level allowed at the RF input, as an overload may lead to hardware damage.

Remote command:
`INPut<ip>:ATTenuation` on page 227
`INPut<ip>:ATTenuation:AUTO` on page 228

Using Electronic Attenuation
If the (optional) Electronic Attenuation hardware is installed on the R&S FPS, you can also activate an electronic attenuator.

In "Auto" mode, the settings are defined automatically; in "Manual" mode, you can define the mechanical and electronic attenuation separately.

Note: Electronic attenuation is not available for stop frequencies (or center frequencies in zero span) above 7 GHz.
In "Auto" mode, RF attenuation is provided by the electronic attenuator as much as possible to reduce the amount of mechanical switching required. Mechanical attenuation may provide a better signal-to-noise ratio, however.

When you switch off electronic attenuation, the RF attenuation is automatically set to the same mode (auto/manual) as the electronic attenuation was set to. Thus, the RF attenuation can be set to automatic mode, and the full attenuation is provided by the mechanical attenuator, if possible.

The electronic attenuation can be varied in 1 dB steps. If the electronic attenuation is on, the mechanical attenuation can be varied in 5 dB steps. Other entries are rounded to the next lower integer value.

If the defined reference level cannot be set for the given attenuation, the reference level is adjusted accordingly and the warning "limit reached" is displayed in the status bar.

Remote command:
`INPut<ip>:EATT:STATe` on page 229
`INPut<ip>:EATT:AUTO` on page 228
`INPut<ip>:EATT` on page 228

4.4.4 Configuring Outputs

Access: "Overview" > "Input / Output" > "Output"

The "Output" tab of the "Input / Output" dialog box contains settings to configure the various signal outputs available on the R&S FPS.

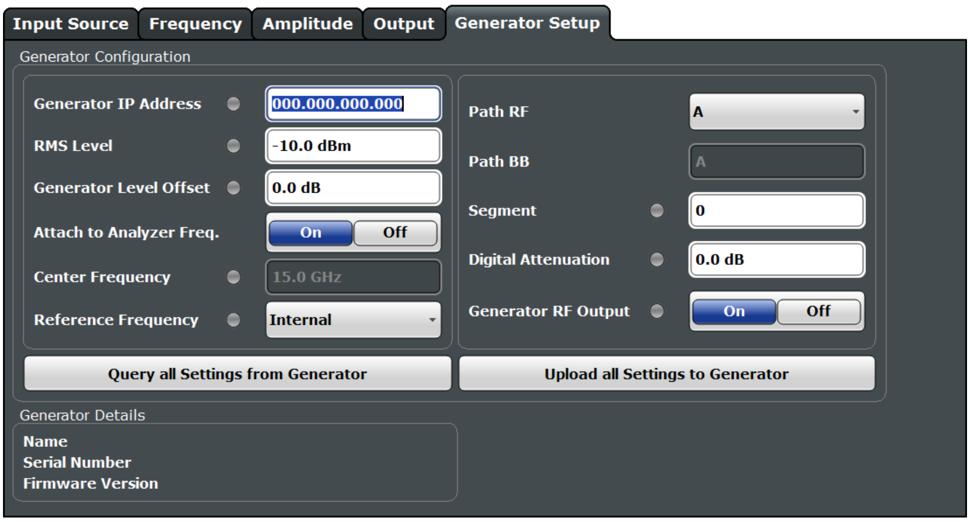
The functionality is the same as in the spectrum application. For more information about the output functions, refer to the R&S FPS user manual.

4.4.5 Controlling a Signal Generator

Access: "Overview" > "Input / Output" > "Generator Setup"

The "Generator Setup" tab of the "Input / Output" dialog box contains settings to control the signal generator from within the R&S FPS-K18. A remote control connection between the R&S FPS and the signal generator has to be established to be able to do so.

Because a signal generator is (mostly) mandatory in the test setup, these features make measurement configuration as easy as possible. This way, you can control both analyzer and generator from within the application without having to operate the two instruments to configure the measurement.



The remote commands required to configure the generator are described in Chapter 6.6.5, "Controlling a Signal Generator", on page 230.

State of operation

Most settings have an LED that shows the state of the corresponding setting on the signal generator.

The LED is either gray, green or red:

- Grey LED
  Configuration state unknown (for example when you have not yet started the transmission).
- Green LED
  Configuration has been successful. Generator has been configured correctly.
- Red LED
  Configuration has not been successful.
  Check if the connection between analyzer and generator has been established or if the IP address has been stated correctly.

Generator details

The "Generator Details" contain information about the connected signal generator, like the software version or the serial number of the generator.

Updating generator settings

When you change the generator level or frequency in this dialog, the application automatically updates those settings on the generator.

When you use the "Upload All Settings To Generator" button, you can force an update of all generator settings available in this dialog box. Useful when you change the level or frequency on the generator itself. In that case, those settings remain the same in the R&S FPS-K18. To restore the original settings defined within the R&S FPS-K18, use that button to restore the generator settings.

Remote command:

CONFigure:GENerator:SETTings:UPDate on page 238

Querying generator settings

Similarly, you can transfer the current generator configuration into the amplifier application with the "Query All Settings From Generator" button.

Note that the center frequency is not updated when you attach the generator frequency to that of the R&S FPS.

Remote command:

CONFigure:SETTings on page 239

Generator Control State

Turns the communication with a connected signal generator on and off.

When you turn off the generator control, the connection between R&S FPS and generator is closed. All settings related to the generator connection (level, frequency, IP address etc.) become unavailable.

Remote command:
CONFigure:GENerator:CONTrol[:STATe] on page 231

IP Address

Opens a dialog box to configure the network properties of the signal generator.

You can connect to the generator either by entering its IP address ("123" button), or its computer name ("ABC" button).

If you are not sure about the IP address or computer name of your generator, check its user interface or kindly ask your IT administrator to provide them.

After you have entered IP address or computer name, use "Connect" to establish the connection. The R&S FPS shows if the connection state, and, if the connection was successful, the connected generator type.

Remote command:
CONFigure:GENerator:IPConnection:ADDRess on page 234
CONFigure:GENerator:IPConnection:LEDState? on page 234

RMS Level

Defines the RMS level of the signal that is generated.

When you define the RMS level here, the signal generator is automatically configured to that level.

In addition, you can define a level offset (for example to take external attenuation into account). Note that the level offset is a purely mathematical value and does not change the actual level of the signal at the RF output.

The level offset takes level offsets into account that occur before the signal has passed through the DUT (usually an amplifier). For level offsets occurring after the DUT, define a level offset in the "Amplitude" menu of the signal analyzer.

You can also define a Digital Attenuation that you can use for fast output level changes.

NOTICE! Risk of damage to the DUT.
RMS levels that are too high can damage or destroy the DUT.

Make sure to keep an eye on the RMS level, especially when defining a level offset. A level offset changes the displayed value of the RMS level, but not the real RMS level.

Displayed RMS level = real RMS level + level offset

Thus, the actual RMS level can be higher than the displayed level.

Note: Always change the generator level from within the R&S FPS-K18 user interface and thus synchronize the levels of both instruments.
If you change the generator level on the signal generator, the R&S FPS-K18 does not synchronize the levels and measurement results are going to be invalid.

Remote command:
RMS level: `CONFigure:GENerator:POWer:LEVel` on page 235
`CONFigure:GENerator:POWer:LEVel:LEDState?` on page 236
Level offset: `CONFigure:GENerator:POWer:LEVel:OFFSet` on page 236
`CONFigure:GENerator:POWer:LEVel:OFFSet:LEDState?` on page 236

Maximum DUT Input Level
Defines the maximum level that the generated signal can have. Selecting a higher level is not possible.

Defining a maximum output level is useful if you are measuring sensitive DUTs.

Remote command:
`CONFigure:GENerator:DUT:INPut:MAXimum:POWer` on page 231
`CONFigure:GENerator:DUT:INPut:MAXimum:POWer:LEDState?` on page 231

Attach to Analyzer Frequency
Turns synchronization of the analyzer and generator frequency on and off.

When you turn on this feature, changing the frequency on the analyzer automatically adjusts the frequency on the generator.

Remote command:
`CONFigure:GENerator:FREQuency:CENTer:SYNC[:STATe]` on page 233

Center Frequency
Defines the frequency of the signal that the generator transmits.

When you turn on Attach to Analyzer Frequency, any changes you make to the generator frequency are also adjusted on the analyzer.

Remote command:
`CONFigure:GENerator:FREQuency:CENTer` on page 232
`CONFigure:GENerator:FREQuency:CENTer:LEDState?` on page 233

Reference Frequency
Selects the source of the generator reference frequency.

The internal reference is that of the signal generator itself. When you select an external reference, you can use another frequency reference, for example that of the R&S FPS.

Remote command:
`CONFigure:GENerator:EXTernal:ROSCillator` on page 232
`CONFigure:GENerator:EXTernal:ROSCillator:LEDState?` on page 232

Path RF / BB
Selects the RF signal path of the generator that is used for signal generation.

Remote command:
RF path: `CONFigure:GENerator:TARGet:PATH:RF` on page 239
BB path: `CONFigure:GENerator:TARGet:PATH:BB?` on page 238

Segment
If you are using a waveform file that contains several different waveforms, you have to select the segment to transfer to the signal generator.

Note that the segment that you have selected in the "Generator Setup" has to match the segment selected for the reference signal, regarding the signal characteristics.

Remote command:
`CONFigure:GENerator:SEGMent` on page 237
`CONFigure:GENerator:SEGMent:LEDState?` on page 238

Digital Attenuation
Attenuates or amplifies the internal, digitally modulated I/Q signal on the signal generator. The level of the RF signal is thus adjusted accordingly.

Digital attenuation allows very fast level changes of the internal I/Q signals.

Note that digital attenuation only has an effect on the RF output level if the internal I/Q modulator of the generator is active.

Remote command:
`CONFigure:GENerator:POWer:LEVel:ATTenuation` on page 235

RF Output
Turns the RF output on the connected signal generator on and off.

When you turn off the RF output, the generator does not feed a signal into the connected DUT.

Remote command:
`CONFigure:GENerator:RFOutput[:STATe]` on page 237
`CONFigure:GENerator:RFOutput:LEDState?` on page 237

Settling Delay

The "Settling Delay" defines a time period between the time a parameter changes on the generator and the start of the next measurement. The R&S FPS automatically waits for the defined time period whenever one of the relevant generator settings has been changed.

Defining a delay time is especially useful for measurements that automatically change generator settings (for example the parameter sweep). The delay time considers the settling time of the generator's hardware components between individual measurements.

Remote command:
`CONFigure:DUT:STIMe` on page 231

4.4.6 Reference: I/Q File Input

4.4.6.1 Basics on Input from I/Q Data Files

The I/Q data to be evaluated in a particular R&S FPS application can not only be captured by the application itself, it can also be loaded from a file, provided it has the correct format. The file is then used as the input source for the application.

For example, you can capture I/Q data using the I/Q Analyzer application, store it to a file, and then analyze the signal parameters for that data later using the Pulse application (if available).

An application note on converting Rohde & Schwarz I/Q data files is available from the Rohde & Schwarz website:

1EF85: Converting R&S I/Q data files

As opposed to importing data from an I/Q data file using the import functions provided by some R&S FPS applications, the data is not only stored temporarily in the capture buffer, where it overwrites the current measurement data and is in turn overwritten by a new measurement. Instead, the stored I/Q data remains available as input for any number of subsequent measurements. Furthermore, the (temporary) data import requires the current measurement settings in the current application to match the settings that were applied when the measurement results were stored (possibly in a different application). When the data is used as an input source, however, the data acquisition settings in the current application (attenuation, center frequency, measurement bandwidth, sample rate) can be ignored. As a result, these settings cannot be changed in the current application. Only the measurement time can be decreased, in order to perform measurements on an extract of the available data (from the beginning of the file) only.

When using input from an I/Q data file, the [RUN SINGLE] function starts a single measurement (i.e. analysis) of the stored I/Q data, while the [RUN CONT] function repeatedly analyzes the same data from the file.

Sample iq.tar files

If you have the optional R&S FPS VSA application (R&S FPS-K70), some sample `iq.tar` files are provided in the `C:/R_S/Instr/user/vsa/DemoSignals` directory on the R&S FPS.

Pre-trigger and post-trigger samples

In applications that use pre-triggers or post-triggers, if no pre-trigger or post-trigger samples are specified in the I/Q data file, or too few trigger samples are provided to satisfy the requirements of the application, the missing pre- or post-trigger values are filled up with zeros. Superfluous samples in the file are dropped, if necessary. For pre-trigger samples, values are filled up or omitted at the beginning of the capture buffer, for post-trigger samples, values are filled up or omitted at the end of the capture buffer.

4.4.6.2 I/Q Data File Format (iq-tar)

I/Q data is packed in a file with the extension `.iq.tar`. An `iq-tar` file contains I/Q data in binary format together with meta information that describes the nature and the source of data, e.g. the sample rate. The objective of the `iq-tar` file format is to separate I/Q data from the meta information while still having both inside one file. In addition, the file format allows you to include user-specific data and to preview the I/Q data in a web browser (not supported by all web browsers).

The `iq-tar` container packs several files into a single `.tar` archive file. Files in `.tar` format can be unpacked using standard archive tools (see http://en.wikipedia.org/wiki/Comparison_of_file_archivers) available for most operating systems. The advantage of `.tar` files is that the archived files inside the `.tar` file are not changed (not compressed) and thus it is possible to read the I/Q data directly within the archive without the need to unpack (untar) the `.tar` file first.

An application note on converting Rohde & Schwarz I/Q data files is available from the Rohde & Schwarz website:

1EF85: Converting R&S I/Q data files

Contained files

An `iq-tar` file must contain the following files:

- **I/Q parameter XML file**, e.g. `xyz.xml`
  Contains meta information about the I/Q data (e.g. sample rate). The filename can be defined freely, but there must be only one single I/Q parameter XML file inside an `iq-tar` file.
- **I/Q data binary file**, e.g. `xyz.complex.float32`
  Contains the binary I/Q data of all channels. There must be only one single I/Q data binary file inside an `iq-tar` file.

Optionally, an `iq-tar` file can contain the following file:

- **I/Q preview XSLT file**, e.g. `open_IqTar_xml_file_in_web_browser.xslt`

Contains a stylesheet to display the I/Q parameter XML file and a preview of the I/Q data in a web browser (not supported by all web browsers).
A sample stylesheet is available at http://www.rohde-schwarz.com/file/open_IqTar_xml_file_in_web_browser.xslt.

I/Q Parameter XML File Specification

The content of the I/Q parameter XML file must comply with the XML schema `RsIqTar.xsd` available at: http://www.rohde-schwarz.com/file/RsIqTar.xsd.

In particular, the order of the XML elements must be respected, i.e. `iq-tar` uses an "ordered XML schema". For your own implementation of the `iq-tar` file format make sure to validate your XML file against the given schema.

The following example shows an I/Q parameter XML file. The XML elements and attributes are explained in the following sections.

Sample I/Q parameter XML file: xyz.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl"
href="open_IqTar_xml_file_in_web_browser.xslt"?>
<RS_IQ_TAR_FileFormat fileFormatVersion="1"
xsi:noNamespaceSchemaLocation="RsIqTar.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Name>R&S FPS</Name>
  <Comment>Here is a comment</Comment>
  <DateTime>2011-01-24T14:02:49</DateTime>
  <Samples>68751</Samples>
  <Clock unit="Hz">6.5e+006</Clock>
  <Format>complex</Format>
  <DataType>float32</DataType>
  <ScalingFactor unit="V">1</ScalingFactor>
  <NumberOfChannels>1</NumberOfChannels>
<DataFilename>xyz.complex.float32</DataFilename>
<UserData>
  <UserDefinedElement>Example</UserDefinedElement>
</UserData>
  <PreviewData>...</PreviewData>
</RS_IQ_TAR_FileFormat>
```

Minimum Data Elements

The following information is always provided by an `iq-tar` file export from the R&S FPS. If not specified otherwise, it must be available in all `iq-tar` files used to import data to the R&S FPS.

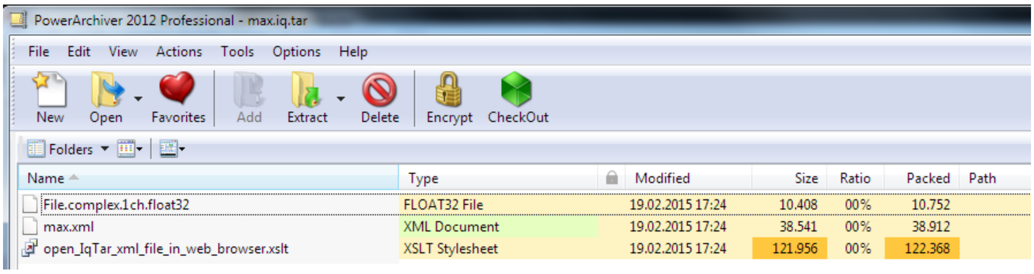
Element	Possible Values	Description
RS_IQ_TAR_FileFormat	-	The root element of the XML file. It must contain the attribute `fileFormatVersion` that contains the number of the file format definition.
Name	string	Optional: describes the device or application that created the file.
Comment	string	Optional: contains text that further describes the contents of the file.
DateTime	yyyy-mm-ddThh:mm:ss	Contains the date and time of the creation of the file. Its type is `xs:dateTime` (see `RsIqTar.xsd`).
Ch<n>_Samples	integer	Contains the number of samples of the I/Q data. For multi-channel signals all channels have the same number of samples. One sample can be: • A complex number represented as a pair of I and Q values • A complex number represented as a pair of magnitude and phase values • A real number represented as a single real value See also `Format` element.
Ch<n>_Clock[Hz]	double	Contains the clock frequency in Hz, i.e. the sample rate of the I/Q data. A signal generator typically outputs the I/Q data at a rate that equals the clock frequency. If the I/Q data was captured with a signal analyzer, the signal analyzer used the clock frequency as the sample rate. The attribute `unit` must be set to "Hz".
Format	complex | real | polar	Specifies how the binary data is saved in the I/Q data binary file (see `DataFilename` element). Every sample must be in the same format. The format can be one of the following: • `complex`: Complex number in cartesian format, i.e. I and Q values interleaved. I and Q are unitless • `real`: Real number (unitless) • `polar`: Complex number in polar format, i.e. magnitude (unitless) and phase (rad) values interleaved. Requires `DataType` = `float32` or `float64`

Element	Possible Values	Description
DataType	int8 | int16 | int32 | float32 | float64	Specifies the binary format used for samples in the I/Q data binary file (see `DataFilename` element and "I/Q Data Binary File" on page 62). The following data types are allowed: • `int8`: 8 bit signed integer data • `int16`: 16 bit signed integer data • `int32`: 32 bit signed integer data • `float32`: 32 bit floating point data (IEEE 754) • `float64`: 64 bit floating point data (IEEE 754)
ScalingFactor	double	Optional: describes how the binary data can be transformed into values in the unit Volt. The binary I/Q data itself has no unit. To get an I/Q sample in the unit Volt the saved samples have to be multiplied by the value of the `ScalingFactor`. For polar data only the magnitude value has to be multiplied. For multi-channel signals the `ScalingFactor` must be applied to all channels. The attribute `unit` must be set to "`V`". The `ScalingFactor` must be > 0. If the `ScalingFactor` element is not defined, a value of 1 V is assumed.
NumberOfChannels	integer	Optional: specifies the number of channels, e.g. of a MIMO signal, contained in the I/Q data binary file. For multi-channels, the I/Q samples of the channels are expected to be interleaved within the I/Q data file (see "I/Q Data Binary File" on page 62). If the `NumberOfChannels` element is not defined, one channel is assumed.
DataFilename	It is recommended that the filename uses the following convention: <xyz>.<Format>.<Channels>ch.<Type> • <xyz> = a valid Windows file name • <Format> = complex, polar or real (see `Format` element) • <Channels> = Number of channels (see `NumberOfChannels` element) • <Type> = float32, float64, int8, int16, int32 or int64 (see `DataType` element)	Contains the filename of the I/Q data binary file that is part of the `iq-tar` file. Examples: • xyz.complex.1ch.float32 • xyz.polar.1ch.float64 • xyz.real.1ch.int16 • xyz.complex.16ch.int8

Element	**Possible Values**	**Description**
UserData	xml	Optional: contains user, application or device-specific XML data which is not part of the `iq-tar` specification. This element can be used to store additional information, e.g. the hardware configuration. User data must be valid XML content.
PreviewData	xml	Optional: contains further XML elements that provide a preview of the I/Q data. The preview data is determined by the routine that saves an `iq-tar` file (e.g. R&S FPS). For the definition of this element refer to the `RsIqTar.xsd` schema. Note that the preview can be only displayed by current web browsers that have JavaScript enabled and if the XSLT stylesheet `open_IqTar_xml_file_in_web_browser.xslt` is available.

Example

The following example demonstrates the XML description inside the `iq-tar` file. Note that this preview is not supported by all web browsers.

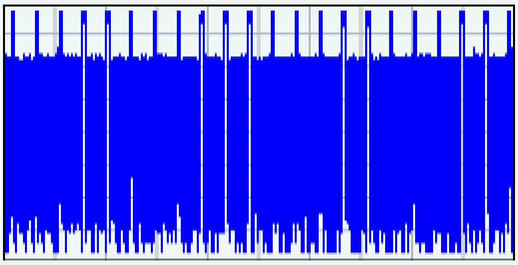
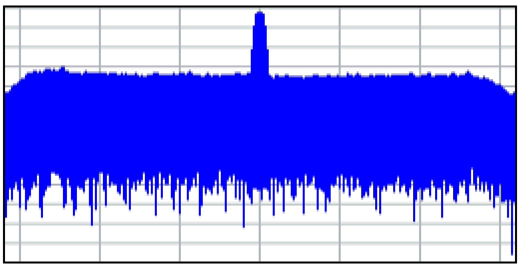
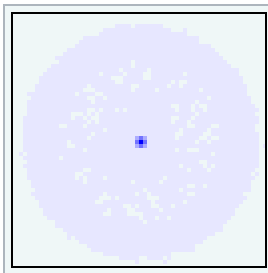


Open the xml file in a web browser, e.g. Microsoft Internet Explorer. If the stylesheet `open_IqTar_xml_file_in_web_browser.xslt` is in the same directory, the web browser displays the xml file in a readable format.

C:\temp\max.xml max.xml

max.xml (of .iq.tar file)

Description	
Saved by	VSE_1.10
Date & Time	2014-11-24 14:34:06
Sample rate	32 MHz
Number of samples	3200300
Duration of signal	100.009 ms
Data format	complex, float32
Data filename	File.complex.1ch.float32
Scaling factor	1 V

IQ Analyzer	
Power vs time y-axis: 10 dB /div x-axis: 10 ms /div	
Spectrum y-axis: 10 dB /div x-axis: 5 MHz /div	
I/Q	

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="open_IqTar_xml_file_in_web_browser.xslt"?>
<RS_IQ_TAR_FileFormat fileFormatVersion="1" xsi:noNamespaceSchemaLocation=
"http://www.rohde-schwarz.com/file/RsIqTar.xsd" xmlns:xsi=
"http://www.w3.org/2001/XMLSchema-instance">
  <Name>VSE_1.10a 29 Beta</Name>
  <Comment></Comment>
  <DateTime>2015-02-19T15:24:58</DateTime>
  <Samples>1301</Samples>
  <Clock unit="Hz">32000000</Clock>
  <Format>complex</Format>
  <DataType>float32</DataType>
  <ScalingFactor unit="V">1</ScalingFactor>
  <NumberOfChannels>1</NumberOfChannels>
  <DataFilename>File.complex.1ch.float32</DataFilename>

<UserData>
    <RohdeSchwarz>
      <DataImportExport_MandatoryData>
        <ChannelNames>
          <ChannelName>IQ Analyzer</ChannelName>
        </ChannelNames>
        <CenterFrequency unit="Hz">0</CenterFrequency>
      </DataImportExport_MandatoryData>
      <DataImportExport_OptionalData>
        <Key name="Ch1_NumberOfPostSamples">150</Key>
        <Key name="Ch1_NumberOfPreSamples">150</Key>
      </DataImportExport_OptionalData>
    </RohdeSchwarz>
  </UserData>

</RS_IQ_TAR_FileFormat>
```

Example: ScalingFactor

Data stored as `int16` and a desired full scale voltage of 1 V

`ScalingFactor` = 1 V / maximum int16 value = 1 V / 2^{15} = 3.0517578125e-5 V

Scaling Factor	Numerical value	Numerical value x ScalingFactor
Minimum (negative) int16 value	$- 2^{15}$ = - 32768	-1 V
Maximum (positive) int16 value	2^{15}-1= 32767	0.999969482421875 V

I/Q Data Binary File

The I/Q data is saved in binary format according to the format and data type specified in the XML file (see `Format` element and `DataType` element). To allow reading and writing of streamed I/Q data, all data is interleaved, i.e. complex values are interleaved pairs of I and Q values and multi-channel signals contain interleaved (complex) samples for channel 0, channel 1, channel 2 etc. If the `NumberOfChannels` element is not defined, one channel is presumed.

Example: Element order for real data (1 channel)

```
I[0],                        // Real sample 0
I[1],                        // Real sample 1
I[2],                        // Real sample 2
...
```

Example: Element order for complex cartesian data (1 channel)

```
I[0], Q[0],                  // Real and imaginary part of complex sample 0
I[1], Q[1],                  // Real and imaginary part of complex sample 1
I[2], Q[2],                  // Real and imaginary part of complex sample 2
...
```

Example: Element order for complex polar data (1 channel)

```
Mag[0], Phi[0],              // Magnitude and phase part of complex sample 0
Mag[1], Phi[1],              // Magnitude and phase part of complex sample 1
Mag[2], Phi[2],              // Magnitude and phase part of complex sample 2
...
```

Example: Element order for complex cartesian data (3 channels)

Complex data: I[channel no][time index], Q[channel no][time index]

```
I[0][0], Q[0][0],           // Channel 0, Complex sample 0
I[1][0], Q[1][0],           // Channel 1, Complex sample 0
I[2][0], Q[2][0],           // Channel 2, Complex sample 0

I[0][1], Q[0][1],           // Channel 0, Complex sample 1
I[1][1], Q[1][1],           // Channel 1, Complex sample 1
I[2][1], Q[2][1],           // Channel 2, Complex sample 1

I[0][2], Q[0][2],           // Channel 0, Complex sample 2
I[1][2], Q[1][2],           // Channel 1, Complex sample 2
I[2][2], Q[2][2],           // Channel 2, Complex sample 2
...
```

Example: Element order for complex cartesian data (1 channel)

This example demonstrates how to store complex cartesian data in float32 format using MATLAB®.

```
% Save vector of complex cartesian I/Q data, i.e. iqiqiq...
N = 100
iq = randn(1,N)+1j*randn(1,N)
fid = fopen('xyz.complex.float32','w');
for k=1:length(iq)
  fwrite(fid,single(real(iq(k))),'float32');
```

```
  fwrite(fid,single(imag(iq(k))),'float32');
end
fclose(fid)
```

Example: PreviewData in XML

```
<PreviewData>
    <ArrayOfChannel length="1">
      <Channel>
        <PowerVsTime>
          <Min>
            <ArrayOfFloat length="256">
              <float>-134</float>
              <float>-142</float>
              ...
              <float>-140</float>
            </ArrayOfFloat>
          </Min>
          <Max>
            <ArrayOfFloat length="256">
              <float>-70</float>
              <float>-71</float>
              ...
              <float>-69</float>
            </ArrayOfFloat>
          </Max>
        </PowerVsTime>
        <Spectrum>
          <Min>
            <ArrayOfFloat length="256">
              <float>-133</float>
              <float>-111</float>
              ...
              <float>-111</float>
            </ArrayOfFloat>
          </Min>
          <Max>
            <ArrayOfFloat length="256">
              <float>-67</float>
              <float>-69</float>
              ...
              <float>-70</float>
              <float>-69</float>
            </ArrayOfFloat>
          </Max>
        </Spectrum>
        <IQ>
          <Histogram width="64" height="64">0123456789...0</Histogram>
        </IQ>
      </Channel>
```

```
  </ArrayOfChannel>
</PreviewData>
```


4.5 Triggering Measurements

Access: "Overview" > "Trigger"

The R&S FPS-K18 provides functionality to trigger measurements.

The "Trigger" dialog box contains settings to configure triggered measurements.

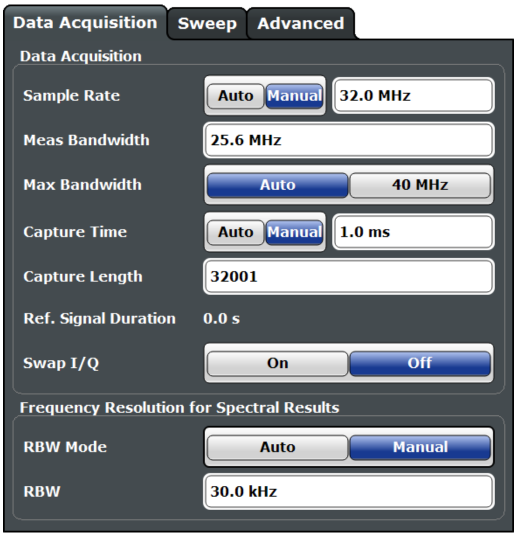
The following trigger sources are supported:
- Free Run
- External
- I/Q Power
- IF Power
- RF Power

The trigger settings are similar to those in the spectrum application. For a comprehensive description of the trigger functionality, refer to the R&S FPS user manual.

4.6 Configuring the Data Capture

Access: "Overview" > "Data Acquisition"

The "Data Acquisition" dialog box contains settings to configure the process of how the application records the signal.



The remote commands required to configure the data capture are described in Chapter 6.6.6, "Configuring the Data Capture", on page 239.

Configuring the measurement bandwidth

The sample rate defined for data acquisition is the sample rate with which the analyzer samples the amplified signal.

The measurement bandwidth defines the flat, usable bandwidth of the final I/Q data.

The application allows you to adjust both values automatically or manually.

Automatic adjustment ← Configuring the measurement bandwidth

When you select automatic adjustment of sample rate and measurement bandwidth, the application selects a bandwidth that is appropriate for the characteristics of the reference signal and adjusts the sample rate accordingly.

For more information about the reference signal, see Chapter 4.3, "Designing a Reference Signal", on page 35.

Remote command:
Mode: `TRACe:IQ:SRATe:AUTO` on page 243

Manual definition ← Configuring the measurement bandwidth

When you define the sample rate and measurement bandwidth manually, you can select values that you are comfortable with. Because the bandwidth is a function of the sample rate (and vice versa), the application adjusts the values when you change either setting.

The following dependencies apply:

- When you change the sample rate, the application updates the bandwidth accordingly (and vice versa). It also adjusts the capture length to the new values. The capture time remains the same.
- When you change the capture time or capture length, the sample rate and bandwidth remain the same.

Remote command:
Sample Rate: `TRACe:IQ:SRATe` on page 242
Bandwidth: `TRACe:IQ:BWIDth` on page 242

Maximum bandwidth ← Configuring the measurement bandwidth

The maximum bandwidth you can use depends on your hardware configuration.

For an overview of available bandwidth extensions, refer to the datasheet.

By default, the application automatically determines the maximum bandwidth. When you select a maximum bandwidth other than "Auto", the bandwidth is restricted to that value. When you select the maximum bandwidth manually, make sure that this bandwidth is suited for the signal you are testing. Otherwise, the signal can be distorted and results are no longer valid.

If you have no bandwidth extension this setting is not available.

For more information about the maximum bandwidth, refer to the user manual of the R&S FPS I/Q Analyzer.

Remote command:
`TRACe:IQ:WBANd[:STATe]` on page 243
`TRACe:IQ:WBANd:MBWidth` on page 243

Configuring the measurement time
The measurement time (or capture time) defines the duration of a measurement in which the required number of samples is collected.

The capture length is the number of samples that are captured during the selected measurement time. The capture length is a function of the sample rate and the capture time.

Automatic adjustment ← Configuring the measurement time
When you select automatic adjustment of capture time, the application selects a capture time that is appropriate for the characteristics of the reference signal.

As orientation, the application shows the length of the reference signal in the corresponding field in the dialog box (→ "Ref Signal Duration").

For more information about the reference signal, see Chapter 4.3, "Designing a Reference Signal", on page 35.

Remote command:
Mode: `[SENSe:]SWEep:TIME:AUTO` on page 242
Reference signal: `[SENSe:]REFSig:TIME?` on page 240

Manual definition ← Configuring the measurement time
When you define the capture length and time manually, you can select values that you are comfortable with.

However, make sure to define a capture time that is greater than the length of the reference signal - otherwise the application is not able to analyze the signal correctly.

The following dependencies apply:
- When you change the capture time, the application updates the capture length accordingly (and vice versa). Sample rate and bandwidth remain the same.
- When you change the sample rate or bandwidth, the application updates the capture length accordingly. The capture time remains the same.

Note that the maximum capture time depends on the current measurement bandwidth.

Remote command:
Time: `[SENSe:]SWEep:TIME` on page 241
Capture length: `[SENSe:]SWEep:LENGth` on page 241

Inverting the I/Q branches
The application allows you to swap the I and Q branches of the signal.

Swapping the branches is useful, for example, when the DUT inverts the real (I) and imaginary (Q) parts of the signal and transfers the signal that way.

Note that the sideband is also inverted when you turn on this feature.

Remote command:
`[SENSe:]SWAPiq` on page 241

Defining the resolution bandwidth for spectrum measurements
The resolution bandwidth (RBW) defines the bandwidth of the resolution filter applied to spectrum measurements (like the Spectrum FFT result).

The "RBW Mode" selects whether the application automatically selects a suitable resolution bandwidth based on the signal you are measuring, or if you define the resolution bandwidth manually. When you select manual definition of the RBW (for example when you want to do a measurement according to a certain telecommunications standard), you can enter the bandwidth in the "RBW" field.

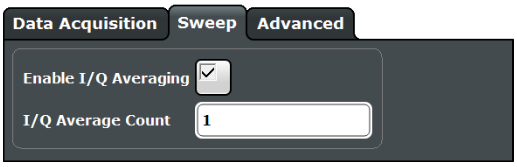
The amplifier measurement application supports any bandwidth between 1 Hz and 10 MHz.

Remote command:
`[SENSe:]BANDwidth[:RESolution]:AUTO` on page 240
`[SENSe:]BANDwidth[:RESolution]` on page 240

4.7 Sweep Configuration

Access: "Overview" > "Data Acquisition" > "Sweep"

The "Sweep" dialog box contains settings to configure the characteristics of a single data recording (a sweep).



The remote commands required to configure the sweep are described in Chapter 6.6.7, "Sweep Configuration", on page 244.

Averaging the I/Q data
Averaging the I/Q data over several data captures can be a useful feature, for example to suppress noise.

Noise suppression is useful in various cases, for example:
- To improve the display of effects like non-linearities or the frequency response.
- To improve the quality of DPD calculation.
- To improve the quality of equalizer training.

When you turn on averaging ("Enable I/Q Averaging"), you can define the number of single data captures the application uses to average the data ("Average Count"). The amount of data captured during a single data capture is defined by the capture length.

By default, the average is reset when the number of single data captures defined by the average count is done.

Example:

When you define an average count of 10, the R&S FPS captures the signal 10 times and then calculates the average over those 10 captures. After that, it resets the results, captures the signal again 10 times and calculates the average etc.

A "Moving Average" on the other hand does not reset the results when the average count is through. Instead, it continues to average the data. The moving average setting only has an effect in continuous sweep mode.

Note that the moving average is automatically disabled during direct DPD calculation.

I/Q averaging is only possible for synchronized parts of the captured signal, because it only makes sense if the same samples in the I/Q data stream are averaged. Therefore, make sure that the measurement is synchronized. Otherwise, the results would be invalid.

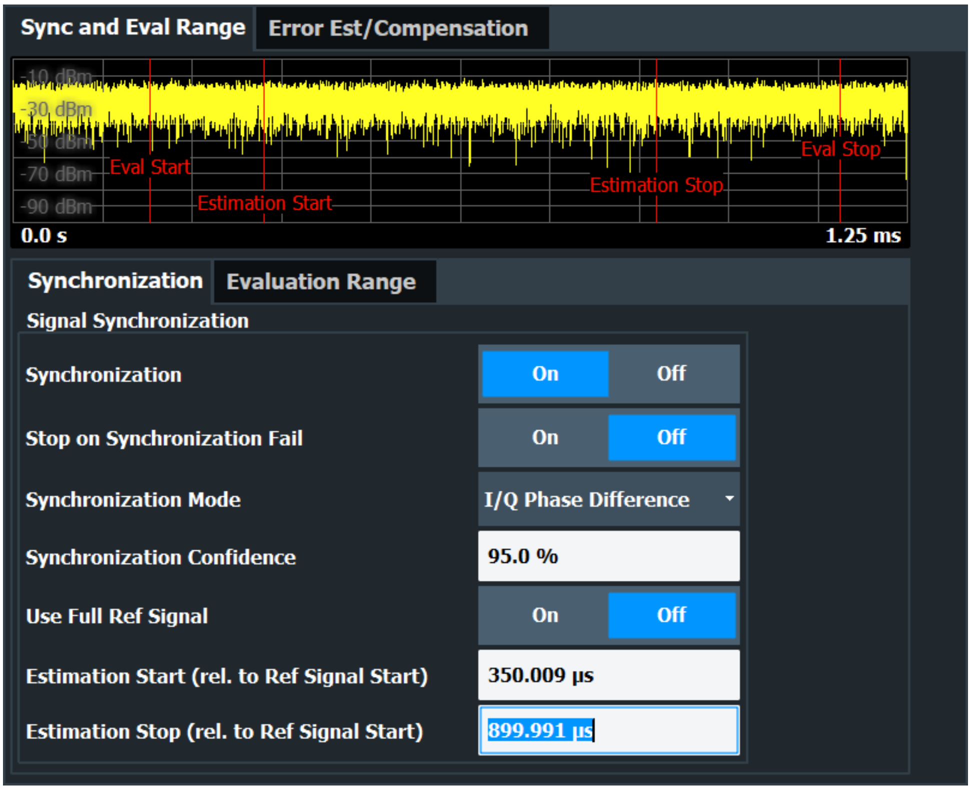
Remote command:
State: `[SENSe:]SWEep:IQAVg[:STATe]` on page 245
Count: `[SENSe:]SWEep:IQAVg:COUNt` on page 244
Moving average: `[SENSe:]SWEep:IQAVg:MAVerage[:STATe]` on page 245

4.8 Synchronizing Measurement Data

Access: "Overview" > "Sync / Error Est / Comp" > "Sync and Eval Range" > "Synchronization"

The application allows you to synchronize the measured signal with the reference signal and provides various features to control synchronization.

Synchronization consists of signal estimation and compensation. After the application has detected the position of the reference signal in the capture buffer, it estimates possible errors in the measured signal (for example the sample error rate or the amplitude droop) by comparing it to the reference signal. The estimated errors can optionally be compensated for.



The remote commands required to configure signal synchronization are described in Chapter 6.6.8, "Synchronizing Measurement Data", on page 246.

Turning synchronization of reference and measured signal on and off

During measurements, the application tries to synchronize the measured signal with the reference signal. When no significant correlation between the measured and reference signal can be found, synchronization fails.

However, you can turn off synchronization if you would like to run unsynchronized measurements. Note however, that the calculation of some results in the result summary requires synchronization. These results cannot be calculated when you turn off synchronization.

When you turn off synchronization, the results are always calculated over the complete capture buffer. When synchronization is on, the results are always calculated over the synchronized data range of the capture buffer. Therefore, the result values can be different for unsynchronized measurements, even if you measure the same signal (the result is still valid and correct, though).

Failed synchronization

When you turn on "Stop on Sync Failed", the application automatically aborts the measurement, in case synchronization fails.

Remote command:
`CONFigure:SYNC:STAT` on page 248
`CONFigure:SYNC:SOFail` on page 248

Selecting the synchronization method

The application allows you to select the method with which the application synchronizes the signals with the "Synchronization Mode" parameter. The following methods are available.

- I/Q Direct
  The I/Q data for the reference signal is directly correlated with the reference and measured signal. The performance of this method degrades in the presence of a frequency offset between the measured and reference signals.
- I/Q Phase Difference
  Correlation on the phase differentiated I/Q data. This method retains phase change information and can handle a frequency offset , but is more sensitive to noise than the "I/Q Direct" method.
- I/Q Magnitude
  Correlation on the magnitude of the I/Q data with no regard for phase information. This method can handle a frequency offset and is less sensitive to noise that the "I/Q Phase Difference" method, but is only useful with amplitude modulated signals.
- Trigger
  It is assumed that the capture is triggered at the start of the reference waveform. Only minimal correlation is performed to account for trigger jitter. This method is the fastest synchronization method.

Remote command:
`CONFigure:SYNC:DOMain` on page 248

Defining a synchronization confidence level

The synchronization confidence level ("Sync Confidence") is a percentage that describes how similar (or correlated) reference and measured signal need to be in order for synchronization to be successful.

A value of 0 % means that synchronization is always successful even if the signals are not correlated at all. However, results that rely on a good synchronization (like the EVM) do contain reasonable values in that case. A value of 100 % means that the signals are identical (in that they are linearly dependent).

The cross-correlation is calculated over all samples in the capture buffer (or the estimation range, if you have defined one).

When the cross-correlation coefficient falls below the confidence level you have defined, synchronization is no longer successful.

Remote command:
`CONFigure:SYNC:CONFidence` on page 247

Defining the estimation range

The estimation range has several effects on the synchronization process.

- It defines which part of the reference signal is used for cross-correlation within the capture buffer in order to align the reference and measured signals.

- It defines which part of the reference signal is used for error estimation.

By default, the application estimates over the complete reference signal. However, you can also estimate over a given range in the capture buffer only. In that case, turn off the "Use Full Ref Signal" feature. When you are not using the full reference signal, the "Eval Start" and "Eval Stop" fields become available. The allowed values are offsets relative to the beginning of the capture buffer (0 s). The highest offset possible depends on the size of the capture buffer.

Defining an estimation range is useful in the following cases.

- If you want to limit the estimation to a specific part of the signal, for example if the signal contains a preamble or midamble.
- If you want to limit the estimation to the ON part of a TDD signal.
- If you want to increase the measurement speed for relatively long signals, for example an LTE signal.

On the downside, limiting the estimation range leads to a higher empirical variance of the results.

In the preview pane displayed in the dialog box, the currently defined estimation range is represented by two red vertical lines.

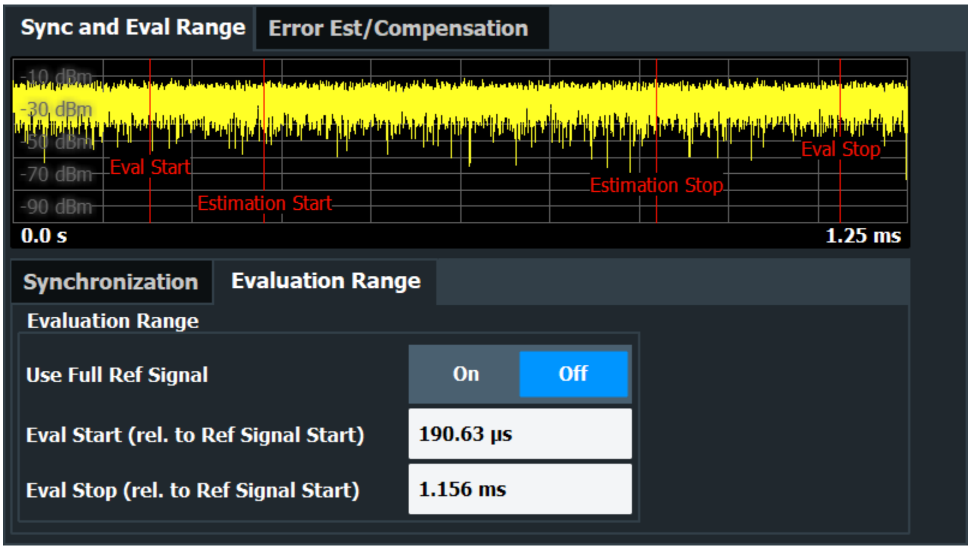
Tip: You can also use the touchscreen to move the lines to a new position in the preview pane. However, this way is not as accurate as entering a number into the input field.

Remote command:
`CONFigure:ESTimation:FULL` on page 246
`CONFigure:ESTimation:STARt` on page 247
`CONFigure:ESTimation:STOP` on page 247

4.9 Evaluating Measurement Data

Access: "Overview" > "Sync / Error Est / Comp" > "Sync and Eval Range" > "Eval Range"

The application allows you to define the time frame in the reference signal used to evaluate and calculate the measurement results.



The remote commands required to configure signal evaluation are described in Chapter 6.6.9, "Defining the Evaluation Range", on page 249.

Defining the evaluation range

The evaluation range defines the data range in the capture buffer over which the application calculates the measurement results.

By default, the application calculates the results over the complete capture buffer. If synchronization has been successful, the application calculates the results over the capture buffer range in which the reference signal has been found. If you have turned off synchronization or if it has not been successful, the complete capture buffer is used to calculate the remaining results.

Example:

The capture buffer is 30 ms long, the reference signal starts at 9 ms and is 10 ms long. When synchronization is successful, the evaluation range starts at 9 ms and ends at 19 ms. If synchronization has been turned off, the evaluation range is the full capture buffer.

However, you can also select a particular data range within the reference signal. In that case, turn off the "Use Full Ref Signal" feature. When it is off, the "Eval Start" and "Eval Stop" fields become available. The allowed values are offsets relative to the beginning of the reference signal (0 s). The highest offset possible depends on the length of the reference signal.

Example:

The situation is as described above (30 ms capture buffer, 10 ms reference signal). Let's say you want to evaluate milliseconds 2 to 6 of the reference signal. In that case, you would have to define a start offset of 11 ms (the reference signal starts at 9 ms, plus the first 2 ms you are not interested in = 11 ms) and a stop offset of 15 ms (9 ms + 6 ms).

In the preview pane displayed in the dialog box, the currently defined evaluation range is represented by two blue vertical lines.

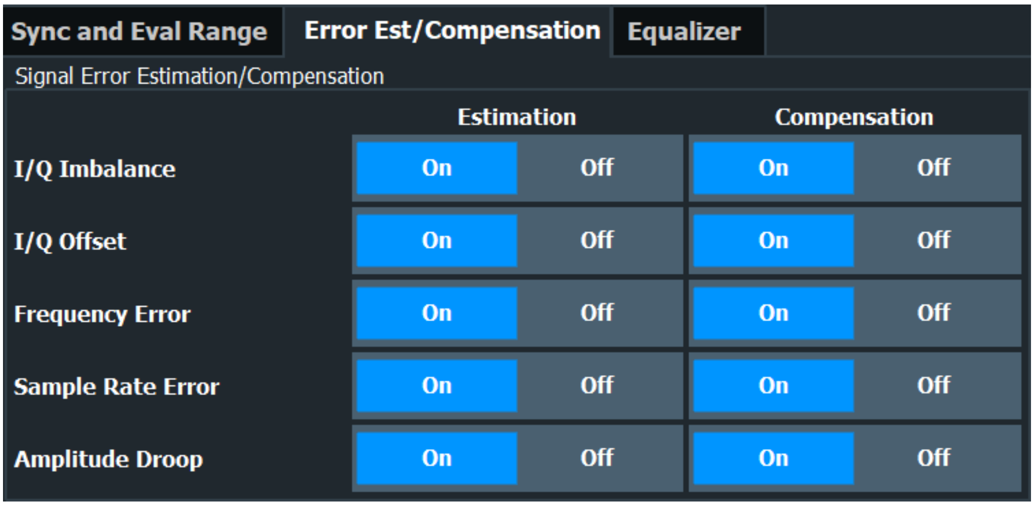
Tip: You can also use the touchscreen to move the lines to a new position in the preview pane. However, this way is not as accurate as entering a number into the input field.

Remote command:
`CONFigure:EVALuation:FULL` on page 249
`CONFigure:EVALuation:STARt` on page 250
`CONFigure:EVALuation:STOP` on page 250

4.10 Estimating and Compensating Signal Errors

Access: "Overview" > "Sync / Error Est / Comp" > "Error Est / Compensation"

The application allows you to estimate possible undesired effects in the signal, and, if there are any, also compensate these effects.



The remote commands required to configure error compensation and equalization are described in Chapter 6.6.10, "Estimating and Compensating Signal Errors", on page 251.

Estimation and compensation

When you turn on error estimation only, the results are not compensated for the corresponding errors.

When you turn on error compensation, the displayed results are also corrected by the estimated errors. Note that in that case, the signal might look better than it actually is.

Compensation without estimation is not possible.

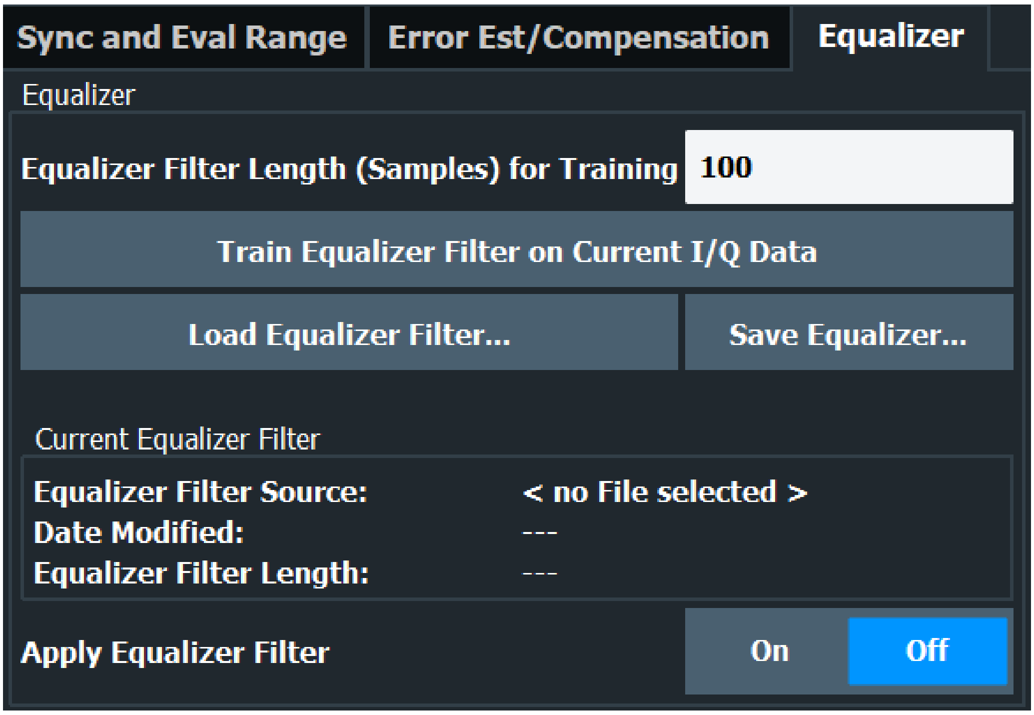
Generally, it is recommended to switch off the estimation of a certain parameter if it is not existent. E.g., if generator and analyzer are frequency locked, it is recommended to switch off the frequency error estimation. Furthermore sample rate error estimation can be switched off if the frequency locked generator is a vector signal generator, i.e. includes the DAC.

You can estimate and compensate the following effects:

- **I/Q Imbalance**: combined effect of amplitude and phase error.
- **I/Q Offset**: shift of the constellation points in a particular direction.
- **Frequency Error**: difference between measured and reference center frequency.
- **Amplitude Droop**: decrease of the signal power over time in the transmitter.
- **Sample Error Rate**: difference between the sample rate of the reference signal and the measured signal.

4.11 Equalizer

In addition, the amplifier application provides equalizer functionality. The equalizer corrects distortions in the frequency characteristics during the transmission of the signal. It can thus help to faithfully reproduce the input signal at the amplifier output.



Using the equalizer
Using the equalizer requires a description of the equalizer filter. You can either train (and save) such a filter automatically with the R&S FPS, or use one that you already have.

Training (or creating) the equalizer filter is a process in which the R&S FPS compares the frequency response of the input and output signal and equalizes potential distortion. The goal is to match the frequency response of the output signal and the input signal. The R&S FPS is able to train the filter based on all samples in the evaluation range.

The "Equalizer Filter Length For Training" property defines the number of FIR filter coefficients to be calculated. A larger number of samples generally yields better results, but takes longer to calculate. After you have defined the filter length (coefficients), you can start the training sequence with the "Train Equalizer Filter on Current I/Q Data" feature. To apply the filter, turn on the equalizer with the "Equalizer State" toggle.

Note that the reference and measured signal need to be synchronized for a successful filter training. Make sure to turn on signal synchronization before you train a filter.

When the filter training is done, you can save the filter in a `csv` or a `fres` file (➙ "Save Equalizer").

For more information about the `fres` file format, refer to the R&S FPS user manual.

If you want to use an equalizer filter that you already have from a previous measurement, you can restore that filter (➙ "Load Equalizer Filter") and apply it without a training sequence.

The dialog box also shows the information about the filter file that is currently in use. This information includes the file name, the date it was modified last and the length of the filter (in samples).

Note: Any equalizer filter is only valid for the sample rate it has been trained for.
If you change the sample rate when an equalizer filter is active, the R&S FPS automatically turns off the equalizer filter. If you still want to use an equalizer filter with the new sample rate, you have to train and apply the equalizer filter again.

Note: An I/Q data export always exports the unequalized (raw) data.
If you want to export the equalized data, you can do so with the following SCPI command.

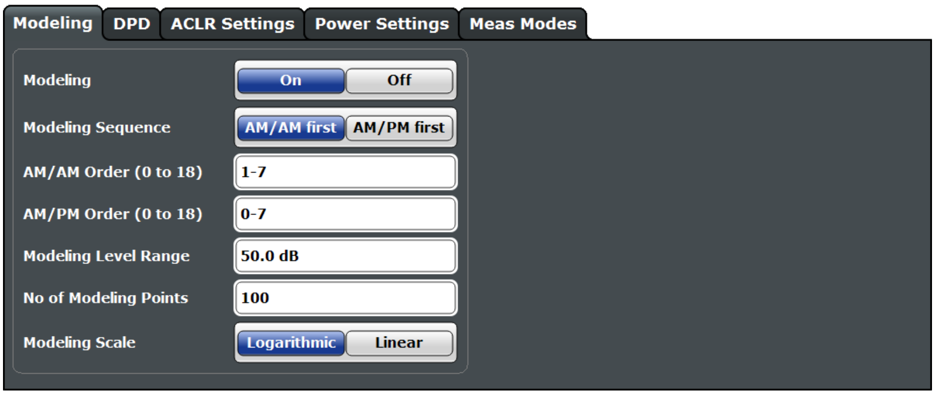
`TRACe:IQ:EQUalized?` on page 207

Remote command:
Filter length: `CONFigure:EQUalizer:FILTer:LENGth` on page 254
Start training: `CONFigure:EQUalizer:TRAin` on page 255
Store filter: `MMEMory:STORe:EQUalizer:FILTer:COEFficient` on page 256
File format: `CONFigure:EQUalizer:FILTer:FILE:FORMat` on page 254
Restore filter: `MMEMory:LOAD:EQUalizer:FILTer:COEFficient` on page 255
Equalizer state: `CONFigure:EQUalizer[:STATe]` on page 255
Manual filter definition: `CONFigure:EQUalizer:FPARameters` on page 254

4.12 Applying System Models

Access: "Overview" > "Measurement" > "Modeling"

A polynomial model describes the characteristics of the DUT based on the input signal and the output signal of the amplifier.



The remote commands required to configure system models are described in Chapter 6.6.11, "Applying a System Model", on page 256.

Turning system modeling on and off

You can use the system modeling functionality to calculate a mathematical model that describes the properties of the DUT.

Using a model is useful to observe and estimate the behavior of the amplifier and, if necessary, adjust the DUT behavior. The application supports memory-free polynomial models to the 18th degree.

The following diagrams contain traces that show the model. These traces are calculated by using the model function on the reference signal.

- AM/AM
- AM/PM

Note that the model traces are also the basis for the DPD functionality available in the R&S FPS-K18.

When the characteristics of the modeled signal match those of the measured signal, the model describes the DUT behavior well. If not, you can try to get a better result by adjusting the model properties.

When you turn on modeling, the application shows an additional trace in the graphical result displays. This trace corresponds to the signal characteristics after the model has been applied to the reference signal.

Selecting the modeling sequence

The modeling sequence selects the sequence in which the models are calculated. The application then either calculates the AM/AM model before calculating the AM/PM model (default), or vice versa.

Remote command:
`CONFigure:MODeling[:STATe]` on page 258
`CONFigure:MODeling:SEQuence` on page 258

Selecting the degree of the polynomial

In addition to the type of curve, you can also select the order of the polynomial model.

The order of the model defines the degree, complexity and number of terms in the polynomial model. In general, a polynomial of the N^{th} degree looks like this:

$y = a_0 + a_1x + a_2x^2 + \dots + a_Nx^N$

The degree of the model is defined by N (as an index or exponent). The higher the order, the more complex the calculation and the longer it takes to calculate the model. Higher models do not necessarily lead to better fitting model curves.

Note that the nonlinear effects consume an additional bandwidth proportional to 2 times the number of odd factors in the polynomial, excluding the linear one.

Example:

If the signal bandwidth is 1 MHz and the highest degree is 5, the bandwidth of the resulting signal is increased by 2 times 2 (because there are the variables a_3 and a_5) times 1 MHz which are 4 MHz. This leads to a total signal bandwidth of 5 MHz (1 MHz + 4 MHz). The configured recording bandwidth must be at least 5 MHz to record all nonlinear effects generated by the DUT.

Tip: To select a specific subset of polynomial degrees you want to apply, you can either:

- Define a range of degrees (e.g. "0 - 5", in that case the application applies all degrees in that range).
- Define a set of individual degrees only (e.g. "1;3;5;7", in that case the application applies those degrees only). Note that the "." key on the front panel draws the ";" character.
- Define a combination of the methods mentioned above (e.g. "1;3;5-7")

Remote command:
AM/AM: `CONFigure:MODeling:AMAM:ORDer` on page 256
AM/PM: `CONFigure:MODeling:AMPM:ORDer` on page 257

Defining the modeling range

The modeling range defines the part of the signal that the model is applied to.

When you limit the level range that the model is applied to, only samples with levels between peak level and "peak level minus modeling level range value" are used during the model calculation. Note that the modeling range is also the range the DPD is applied to.

You can also define a smaller or larger modeling level range. Make sure, however, that the range is large enough not to distort the model.

In addition, you can define the number of points on the curve that the application uses to calculate the model. The selected points are spaced equidistant on a logarithmic scale (an equidistant spacing on a linear scale is also possible if you prefer that). Using fewer modeling points further speeds up measurement times (but can reduce the quality of the model if set too low).

Remote command:
Range: `CONFigure:MODeling:LRANge` on page 257
Points: `CONFigure:MODeling:NPOints` on page 257

Selecting the modeling scale
The input power range is split into several equally spaced subranges (= modeling points) for the calculation of the amplifier model.

With the "Modeling Scale", you can select whether the split is done on a logarithmic or linear basis.

Remote command:
`CONFigure:MODeling:SCALe` on page 257

4.13 Applying Digital Predistortion

Access: "Overview" > "Measurement" > "DPD"

Digital predistortion (DPD) is a method to improve the linearity of an RF power amplifier. Basically, DPD is a set of correction values that is added to the input signal to compensate the non-linearities that occur in the amplifier. The output signal measured by the R&S FPS then shows the corrected amplifier characteristics.

You can compensate non-linearities with the functionality of the amplifier application. The application provides two compensation methods: polynomial DPD and direct DPD.

Note that you can only use one of the two DPD types at any time. When you turn on the polynomial DPD, the R&S FPS automatically turns off the direct DPD and vice versa.

Using the DPD functionality requires a connection to a signal generator. For more information about configuring generators, see Chapter 4.4.5, "Controlling a Signal Generator", on page 50.

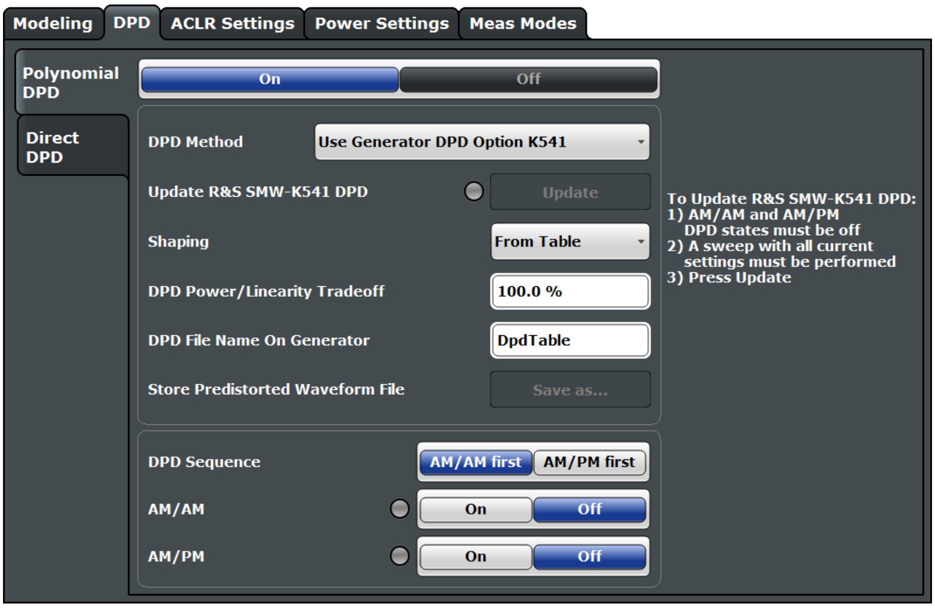
Remote command:

`CONFigure:DDPD[:STATe]` on page 262

4.13.1 Polynomial DPD

For polynomial DPD, the application calculates the correction values based on a polynomial function, whose characteristics you can define with the settings available for the system models. The polynomial DPD approach used by the R&S FPS compensates for AM/AM (amplitude-to-amplitude) distortion and AM/PM (amplitude-to-phase) distortion.

When you apply the DPD, the correction values are applied to the input signal to improve the linearity of the amplifier.



The remote commands required to configure the polynomial DPD are described in Chapter 6.6.12, "Applying Digital Predistortion", on page 258.

Selecting the DPD method

The amplifier application provides a couple of DPD calculation methods.

- "Use Generator DPD Option K541"
  The signal generator corrects the input signal in real time.
  This method requires a Rohde & Schwarz signal generator equipped with option R&S SMx-K541.
  The source of the predistortion values is either a table or a polynomial function.
  After a successful measurement, you can apply the predistortion values that were calculated by the R&S FPS with the "Update" button. (The button is only available when data has been captured on the R&S FPS and synchronization was successful).
  Note that you have to turn on the DPD model in order to make the DPD work.
  As long as you use the same amplifier, the polynomial DPD calculated with this method is valid for all signals that use a similar bandwidth and frequency as the signal it was calculated for.
- "Generate Pre-Distorted Waveform File"
  The R&S FPS applies the correction values taken from the table or polynomial function to each measured sample and generates a waveform file that contains the corrected input signal. For TDD and FDD signals, we recommend that you use the full reference signal to generate the DPD.
  You can start the DPD calculation and transfer the resulting waveform file to the connected generator with the "Generate and Load" button. Successful calculation

and transfer are indicated by a green LED. Note that you have to turn on the DPD model in order to make the DPD work.
Note:
When you use this method, the predistortion information only applies to the currently selected reference signal and generator level. When you change the reference signal or generator level, you have to create a file that applies to the new reference signal.
You can also save the predistorted waveform into a waveform file with the "Store Pre-Distorted Waveform File" feature for later reference.

Remote command:
`CONFigure:DPD:METHod` on page 265

Selecting the DPD shaping method
The application provides several ways for DPD calculation (or shaping).

- "From Table"
Shapes the DPD function based on a table that contains the correction values required to predistort the signal.
The calculation of the table is based on the AM/AM and AM/PM polynomial models.
For more information about the contents and usage of the shaping table, refer to the documentation of the R&S SMW-K541.
You can define a file name for the DPD table in the corresponding field.
- "From Polynomial"
Shapes the DPD function based on a correction polynomial that is calculated out of the model polynomial.
Compared to DPD based on a shaping table, this method does not transfer a list with correction values. Instead, the application transfers the polynomial coefficients of the correction polynomial.
For more information, see Chapter 4.12, "Applying System Models", on page 76.

You can update the DPD shaping on the signal generator comfortably with the "Update" button.

Remote command:
Mode: `CONFigure:DPD:SHAPing:MODE` on page 266
Table name: `CONFigure:DPD:FNAMe` on page 265

Selecting the order of model calculation
The application allows you to compensate for AM/AM distortion, AM/PM distortion or both simultaneously. You can turn correction of the distortion models on and off in the corresponding fields.

If you want to predistort both the AM/AM distortion and the AM/PM distortion simultaneously, you can select the order in which the curves are calculated and applied to the I/Q signal on the R&S SMW.

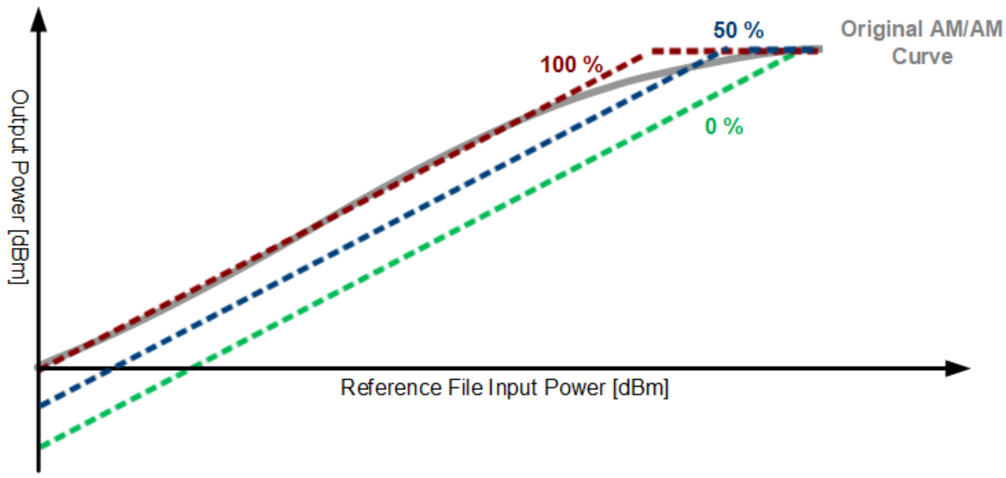
- AM/AM First
Calculates the AM/AM first, then calculates the AM/PM based on the signal that has already been corrected by its AM/AM distortions.
- AM/PM First
Calculates the AM/PM first, then calculates the AM/AM based on the signal that has already been corrected by its AM/PM distortions.

Note: the DPD sequence is displayed by the diagram that is part of the dialog box.

Remote command:
AM/AM state: `CONFigure:DPD:AMAM[:STATe]` on page 262
AM/PM state: `CONFigure:DPD:AMPM[:STATe]` on page 263
Both: `CONFigure:DPD:AMXM[:STATe]` on page 263
Calculation order: `CONFigure:DPD:SEQuence` on page 266

DPD Power / Linearity Tradeoff
The "DPD Power / Linearity Tradeoff" describes the effects of the DPD on the amplifier characteristics.

When you define a tradeoff of 0 %, the DPD aims for the best linearity (green line in the illustration below). When you increase the tradeoff value, the DPD aims for an optimization of the output power at the expense of linearity. In the 100 % case, output power is maximized, whereas linearity is reduced compared to all other cases. The blue line shows the default tradeoff value of 50 %.



Remote command:
Polynomial DPD: `CONFigure:DPD:TRADeoff` on page 266
Direct DPD: `CONFigure:DDPD:TRADeoff` on page 262

4.13.2 Direct DPD (R&S FPS-K18D)

The direct DPD is an iterative process in which the correction values are determined for each sample of the input signal. Compared to the polynomial DPD, the direct DPD is not based on a model. It rather calculates the correction values for each sample directly.

Determining the DPD directly is based on a sequence of individual measurements (iterations). When one iteration is done, the R&S FPS applies the correction values, measures the improved input signal again, applies the correction values etc. This process goes on until the number of iterations you have defined is done. Usually, the predistortion gets better with an increasing number of iterations. On the other hand, increasing the number of iterations also increases the measurement time.

Note that synchronization must be successful in every iteration to get a DPD. The R&S FPS repeats every iteration up to 10 times if synchronization fails. If synchronization was not possible 10 times in a row, the R&S FPS stops the generation of the direct DPD and shows a corresponding message. Reducing the synchronization confidence level can help in that case.

The result of the direct DPD is an I/Q file that contains a predistorted waveform. When you save the I/Q file, you can later play it back on a signal generator.

For TDD and FDD signals, we recommend that you use the full reference signal to generate the DPD.

Further improvement of predistortion

In addition to increasing the number of iterations, it is recommended to apply signal averaging during each iteration. Averaging helps to remove noise from the signal, which in turn improves the quality of the predistortion values.

Without averaging, each iteration consists of a single measurement. When you apply averaging, the number of measurements during each iteration increases, depending on the number of averages you have defined.

The advantage of the direct DPD compared to the polynomial DPD is, that it takes memory effects into account. This, and the fact that it is not based on a model, but corrects each sample individually, makes the direct DPD the superior method to predistort the input signal and determine the ideal DPD effect for your DUT. Note however, that the correction values that have been determined are only applicable to the signal and amplifier you have used. If the signal characteristics change in any way, you have to predistort the signal again.

The direct DPD is especially useful for the following test cases.

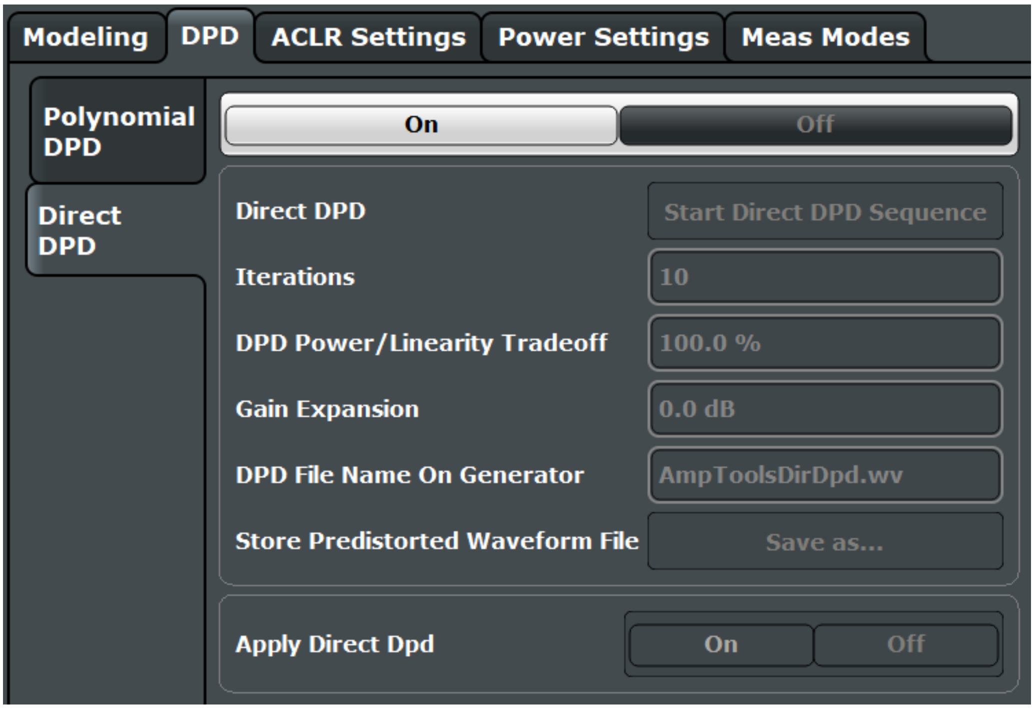
- Determining the best performance of a DUT.
- Removing external effects from the measurement results, for example a preamplifier that should not be considered in the final measurement results.

Moving averages during direct DPD calculation

The moving average is automatically disabled during the direct DPD calculation.

Generator control during direct DPD calculation

When direct DPD is activated, the generator will be prevented from changing its attenuator setting automatically, i.e. it’s being set into mode "Fixed" if it was in "Auto" mode so far. The attenuator mode will be switched back to "Auto" when direct DPD is turned off. If the generator was in "Fixed" or "Manual" mode, the mode is not changed.



The remote commands required to configure the direct DPD are described in Chapter 6.6.12, "Applying Digital Predistortion", on page 258.

Functions available for the direct DPD described elsewhere:

- "DPD Power / Linearity Tradeoff" on page 82

Running a direct DPD sequence

The direct DPD method requires one or more measurements (or iterations) to determine the correction values.

When you select the "Start Direct DPD Sequence" button, the R&S FPS initiates a sequence of measurements during which the DPD is calculated. The number of measurements performed during the sequence depends on the number of "Iterations" you have defined. It is also recommended to average each iteration for further improvement of the quality of the input signal. The "Gain Expansion" defines the increase of input power relative to the peak power value of the reference signal.

You can follow the process of the DPD sequence in the channel bar. The "DPD Count" label shows the current iteration and the complete number of iterations of the DPD sequence. If you are using averaging, the "Count" label shows the process of the current iteration. The first number is the current measurement, the second number the total number of measurements.

When the DPD sequence is done, the R&S FPS stores the predistorted I/Q signal in a waveform file and transfers it to the signal generator. You can change the name of the waveform file in the "DPD File Name on Generator" property. The waveform file is transferred automatically to the generator. It is loaded into the ARB when you turn on the "Apply Direct DPD" property. (Note that when you turn off the direct DPD again, the generator restores the waveform file that was previously used.)

You can also save the waveform file, for example if you want to use it again later, with the "Store Predistorted Waveform File" property.

Note that you can stop a DPD sequence any time through the dialog box shown while the DPD sequence is running.

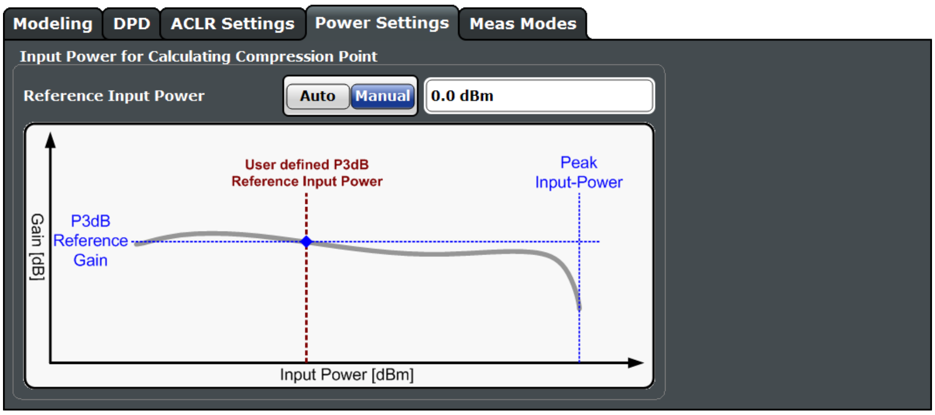
- "Finish": Stops the DPD sequence and keeps the predistorted I/Q data that have already been calculated.
- "Abort": Stops the DPD sequence and discards the predistorted I/Q data that have already been calculated.

Remote command:
Iterations: `CONFigure:DDPD:COUNt` on page 260
Start sequence: `CONFigure:DDPD:STARt` on page 261
Gain expansion: `CONFigure:DDPD:GEXPansion` on page 261
File name: `CONFigure:DDPD:FNAMe` on page 261
Save DPD: `MMEMory:STORe:DDPD` on page 269
Apply DPD: `CONFigure:DDPD:APPLy[:STATe]` on page 259

4.14 Configuring Power Measurements

Access: "Overview" > "Measurement" > "Power Settings"

The Amplifier application features functionality to configure measurements that determine power characteristics of an amplifier.



The remote commands required to configure power measurements are described in Chapter 6.6.14, "Configuring Power Measurements", on page 275.

Configuring compression point calculation
The application evaluates three compression points. The compression points represent the input power where the gain of the amplifier deviates by a certain amount from a reference point on the gain curve. The amount of deviation is either 1 dB, 2 dB or 3 dB.

Because these compression points are relative values, you have to define the reference gain.

There are two ways to get the reference gain: automatically or manually.

If you define the reference gain **manually**, the reference point is the gain at a certain input power (which you can define in the "Reference Input Power" input field).

If you select **automatic** calculation of the reference gain, the reference gain is the average gain that has been measured (the average gain is a result shown in the Numeric Result Summary).

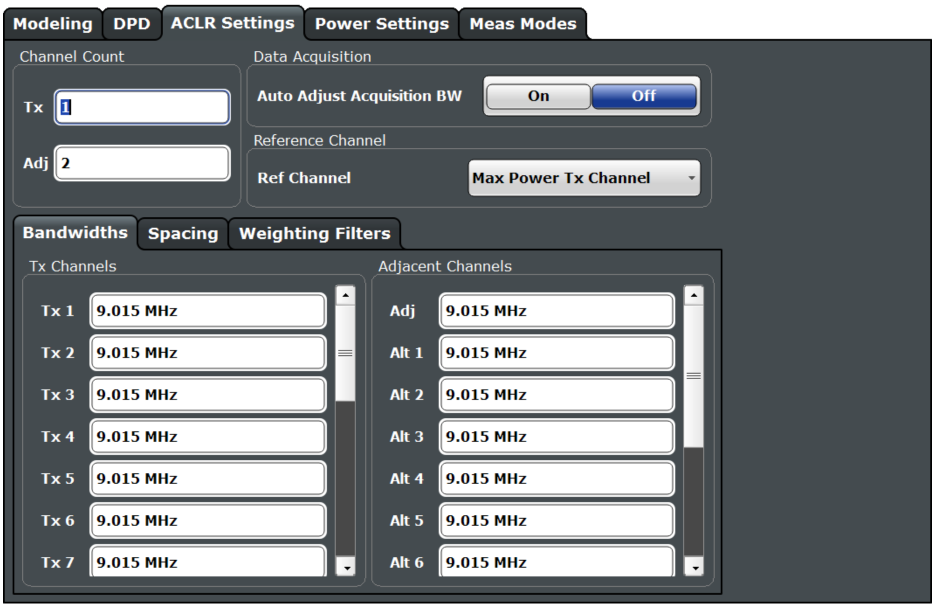
In the Gain Compression result display, the reference point is indicated by a red line.

Remote command:
Method: `CONFigure:POWer:RESult:P3DB[:STATe]` on page 275
Input power: `CONFigure:POWer:RESult:P3DB:REFerence` on page 275

4.15 Configuring Adjacent Channel Leakage Error (ACLR) Measurements

Access: "Overview" > "Measurement" > "ACLR Settings"

The application allows you to define the basic characteristics of the Tx channel and neighboring channels when you perform ACLR measurements.



The remote commands required to configure the ACLR measurements are described in Chapter 6.6.13, "Configuring ACLR Measurements", on page 269.

Number of channels: Tx , Adj

Up to 18 carrier channels and up to 12 adjacent channels can be defined.

Results are provided for the Tx channel and the number of defined adjacent channels *above and below* the Tx channel. If more than one Tx channel is defined, the carrier channel to which the relative adjacent-channel power values should be referenced must be defined (see " Reference Channel " on page 87).

Remote command:
Number of Tx channels:
`[SENSe:]POWer:ACHannel:TXCHannel:COUNt` on page 275
Number of Adjacent channels:
`[SENSe:]POWer:ACHannel:ACPairs` on page 270

Selecting the measurement bandwidth

When you perform an ACLR measurement, it is important to select a measurement bandwidth that is large enough to capture all channels you want to evaluate in the ACLR measurement.

The application provides automatic adjustment of the measurement bandwidth to the bandwidth occupied by all channels evaluated in the ACLR measurement. To do so, turn on the "Auto Adjust Acquisition Bandwidth" function.

Note that you also have to turn on automatic bandwidth selection in the "Data Acquisition" dialog box in order to adjust the measurement bandwidth to the ACLR configuration.

If you define the bandwidth manually, make sure to take one that is large enough to capture all channels. Otherwise, the R&S FPS does not evaluate measurement results. Also make sure that the R&S FPS you are using can actually handle the bandwidth occupied by the transmission and adjacent channels. For larger bandwidths, one of the I/Q bandwidth extensions could be necessary (refer to the datasheet for a complete list of available bandwidth extensions).

Remote command:
`[SENSe:]POWer:ACHannel:AABW` on page 270

Reference Channel

The measured power values in the adjacent channels can be displayed relative to the transmission channel. If more than one Tx channel is defined, define which one is used as a reference channel.

Tx Channel 1	Transmission channel 1 is used. (Not available for MSR ACLR)
Min Power Tx Channel	The transmission channel with the lowest power is used as a reference channel.
Max Power Tx Channel	The transmission channel with the highest power is used as a reference channel (Default).
Lowest & Highest Channel	The outer left-hand transmission channel is the reference channel for the lower adjacent channels, the outer right-hand transmission channel that for the upper adjacent channels.

Remote command:
[SENSe:]POWer:ACHannel:REFerence:TXCHannel:MANual on page 273
[SENSe:]POWer:ACHannel:REFerence:TXCHannel:AUTO on page 273

Channel Bandwidth

The Tx channel bandwidth is normally defined by the transmission standard.

The value entered for any Tx channel is automatically also defined for all subsequent Tx channels. Thus, only enter one value if all Tx channels have the same bandwidth.

The value entered for any ADJ or ALT channel is automatically also defined for all alternate (ALT) channels. Thus, only enter one value if all adjacent channels have the same bandwidth.

Remote command:
[SENSe:]POWer:ACHannel:BANDwidth[:CHANnel<ch>] on page 271
[SENSe:]POWer:ACHannel:BANDwidth:ACHannel on page 271
[SENSe:]POWer:ACHannel:BANDwidth:ALTernate<ch> on page 271

Channel Spacings

Channel spacings are normally defined by the transmission standard but can be changed.

If the spacings are not equal, the channel distribution in relation to the center frequency is as follows:

Odd number of Tx channels	The middle Tx channel is centered to center frequency.
Even number of Tx channels	The two Tx channels in the middle are used to calculate the frequency between those two channels. This frequency is aligned to the center frequency.

The spacings between all Tx channels can be defined individually. When you change the spacing for one channel, the value is automatically also defined for all subsequent Tx channels. This allows you to set up a system with equal Tx channel spacing quickly. For different spacings, set up the channels from top to bottom.

Tx1-2	Spacing between the first and the second carrier
Tx2-3	Spacing between the second and the third carrier
…	…

If you change the adjacent-channel spacing (ADJ), all higher adjacent channel spacings (ALT1, ALT2, …) are multiplied by the same factor (new spacing value/old spacing value). Again, only enter one value for equal channel spacing. For different spacing, configure the spacings from top to bottom.

Remote command:
[SENSe:]POWer:ACHannel:SPACing:CHANnel<ch> on page 274
[SENSe:]POWer:ACHannel:SPACing[:ACHannel] on page 274
[SENSe:]POWer:ACHannel:SPACing:ALTernate<ch> on page 274

Weighting Filters
Weighting filters allow you to determine the influence of individual channels on the total measurement result. For each channel you can activate or deactivate the use of the weighting filter and define an individual weighting factor ("Alpha:" value).

Remote command:
Activating/Deactivating:
[SENSe:]POWer:ACHannel:FILTer[:STATe]:CHANnel<ch> on page 273
[SENSe:]POWer:ACHannel:FILTer[:STATe]:ACHannel on page 272
[SENSe:]POWer:ACHannel:FILTer[:STATe]:ALTernate<ch> on page 273
Alpha value:
[SENSe:]POWer:ACHannel:FILTer:ALPHa:CHANnel<ch> on page 272
[SENSe:]POWer:ACHannel:FILTer:ALPHa:ACHannel on page 272
[SENSe:]POWer:ACHannel:FILTer:ALPHa:ALTernate<ch> on page 272

4.16 Configuring the Parameter Sweep

Access: "Overview" > "Measurements" > "Meas Modes" > "Parameter Sweep"

The parameter sweep is a measurement that allows you to compare a result (that you can select arbitrarily) against two other parameters. The advantage of the parameter sweep is that it controls the signal generator and the analyzer, and automatically changes the signal characteristics (for example the frequency) without you having to do those changes manually. In addition, it combines the results in a single and well arranged diagram and / or numerical result display (→ Parameter Sweep).

Example:

In the default state, the application compares the EVM against the frequency and the generator power.

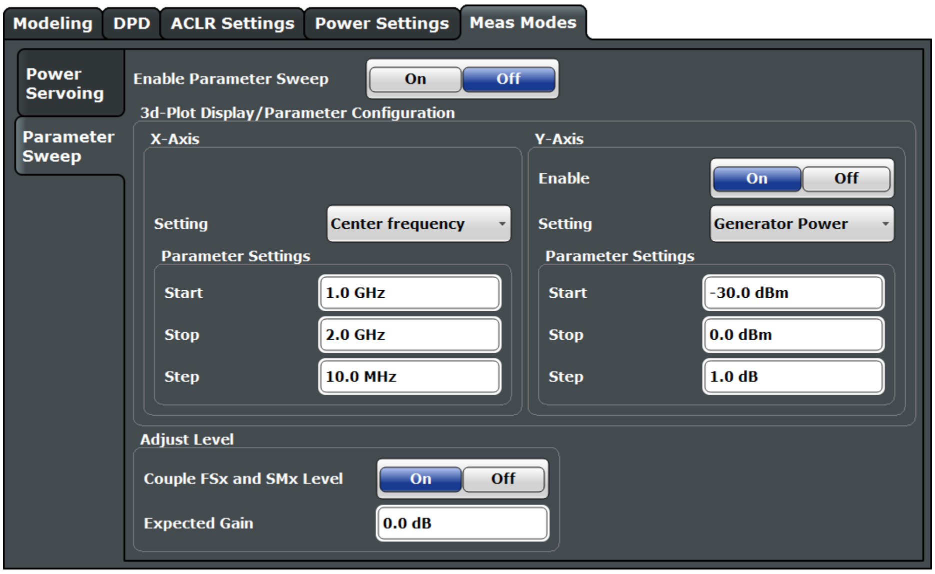
In that case, the R&S FPS first performs a measurement on the first frequency for each generator output level in the defined range. When this measurement is done, the R&S FPS continues to measure all power levels on the second frequency and so on.

Frequency range: 10 MHz to 20 MHz, stepsize 1 MHz. Output level range: -10 dBm to 0 dBm, stepsize: 1 dB.

- 1st measurement: 10 MHz with a generator output level of -10 dBm.
- (...)
- 11th measurement: 10 MHz with a generator output level of 0 dBm.
- 12th measurement: 11 MHz with a generator output level of -10 dBm.
- (...)
- 22nd measurement: 11 MHz with a generator output level of 0 dBm.
- (...)
- nth measurement: 20 MHz with a generator output level of 0 dBm.

The configuration affects the number of measurements that will be performed. The number of measurements in turn has an effect on the overall measurement time of the parameter sweep.

The parameter sweep requires a connection to a signal generator. For more information about configuring generators, see Chapter 4.4.5, "Controlling a Signal Generator", on page 50.



The remote commands required to configure the parameter sweep are described in Chapter 6.6.15, "Configuring Parameter Sweeps", on page 276.

Turning the parameter sweep on and off

Before you can use the parameter sweep functionality, you have to turn it on deliberately.

When you turn it on, the R&S FPS starts the parameter sweep in single sweep mode ([Run Sgl] and [Run Cont] both start the parameter sweep in that case). When the parameter sweep is on, other measurements are not possible, and vice versa.

Turning on the parameter sweep also expands the channel bar by several labels that carry information about the progress of the parameter sweep.

Remote command:
`CONFigure:PSWeep[:STATe]` on page 277

Selecting the data to be evaluated during the parameter sweep

When you are performing a parameter sweep, you can compare an arbitrary result against one or two arbitrary parameters.

Depending on your selection, the R&S FPS changes the values of the selected parameters on the signal generator during the measurement, and calculates the result for each combination of values.

If there is more than one instance of the parameter sweep, the R&S FPS applies the selected parameters to all instances. The displayed results on the other hand, can be different for each instance.

- Center Frequency
  Controls the frequency of the signal generator.
- Generator Power
  Controls the output power of the signal generator.
- Envelope to RF Delay
  Controls the delay between the envelope and the RF signal on the signal generator.
- Envelope Bias
  Controls the envelope bias on the signal generator.

You can define the scope of the measurement by adjusting the start and stop values for both parameters, and assign a certain stepsize. Based on these values, the R&S FPS changes the generator setup after each individual measurement.

The second parameter is not mandatory. You can turn it off with the "Y-Axis Enable" function. In that case, the parameter sweep is represented in a two-dimensional diagram (for example the EVM against the frequency).

Example:

When you define a level range from 0 dBm (start value) to 10 dBm (stop value) with a stepsize of 1 dB, the parameter sweep would perform 11 measurements on a single frequency.

When you also define a frequency range between 10 MHz and 20 MHz, and a stepsize of 1 MHz, the total number of measurements would be 121: 11 power level measurements on each of the 11 frequencies.

Remote command:
Chapter 6.6.15, "Configuring Parameter Sweeps", on page 276

Synchronizing the levels of signal generator and analyzer

When you sweep the output level of the generator, make sure to synchronize the reference level of the analyzer and the RMS level of the generator to avoid damage to the RF input of the analyzer (→ "Couple FSx and SMx Level"). When you do so, the application automatically matches the reference level of the analyzer to the output level of the generator.

For sensitive DUTs, you can define maximum output level that is not exceeded during the parameter sweep.

Note that it is mandatory to define the "Expected Gain" of the DUT. Otherwise, the synchronization between the levels can fail or lead to invalid results.

NOTICE! Risk of damage to the RF input of the analyzer.
Make sure to define the correct "Expected Gain". Otherwise, the R&S FPS does not consider the gain of the amplifier during the level changes on signal analyzer and generator, which in turn can lead to a high-level signal damaging or destroying the RF input mixer of the analyzer.

With a correct "Expected Gain" value, however, the application is able to attenuate the signal accordingly.

Remote command:
Synchronization state: `CONFigure:PSWeep:ADJust:LEVel[:STATe]` on page 276
Expected gain: `CONFigure:PSWeep:EXPected:GAIN` on page 276

5 Analysis

The amplifier application provides several tools to get more information about the results.

Most of these tools work similar to those available in the spectrum application. For more information about these tools, refer to the R&S FPS user manual.

5.1 Configuring Traces

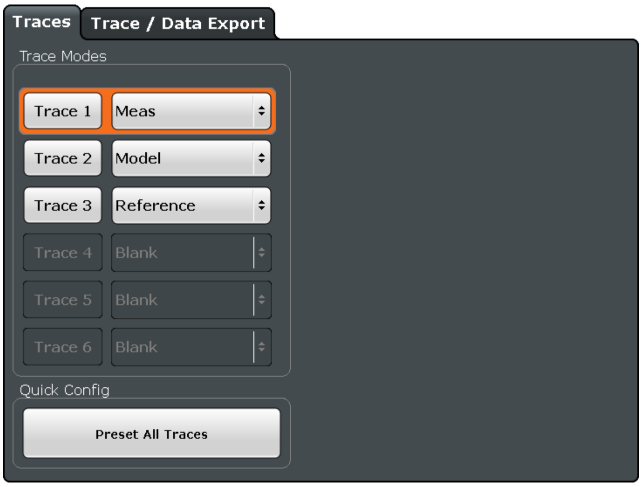
The amplifier application provides several tools to configure and evaluate traces.

5.1.1 Selecting the Trace Information

Access: [TRACE] > "Trace Config" > "Traces"

Each result display contains one or several traces specific to the corresponding result type.

The number of traces available for each result display and the information these traces provide are described in Chapter 3, "Measurements and Result Displays", on page 11.



Trace Mode
Defines the update mode for subsequent traces.

Clear Write	Overwrite mode (default): the trace is overwritten by each measurement.
Max Hold	The maximum value is determined over several measurements and displayed. The R&S FPS saves each trace point in the trace memory only if the new value is greater than the previous one.
Min Hold	The minimum value is determined from several measurements and displayed. The R&S FPS saves each trace point in the trace memory only if the new value is lower than the previous one.
Average	The average is formed over several measurements.
View	The current contents of the trace memory are frozen and displayed.
Blank	Removes the selected trace from the display.

Remote command:
`DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:MODE` on page 280

Detector
Defines the trace detector to be used for trace analysis.

Positive Peak	The positive detector displays the maximum level that has been detected during the measurement.
Negative Peak	The negative peak detector displays the minimum level that has been detected during the measurement.
Average	The average detector displays an RMS average (linear and quadratic) for most traces including EVM. Only for VCC/ICC traces (linear averaged) the average voltages and currents are displayed.
Off	No specific detector is active and all values are recorded.

Remote command:
`[SENSe:][WINDow<n>:]DETector<t>[:FUNCtion]` on page 284

Result Type
Defines the result type to be used for trace analysis.

IdealLine	Displays a line that equals to a perfect linear device for "AM/AM", "AM/PM" and "Gain Compression" traces.
Meas	Displays the measured signal.
Model	Displays a modeled signal for "AM/AM" and "AM/PM" traces.
Reference	Displays the reference signal for "FFT Spectrum" traces.

Remote command:
`DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:RESult` on page 283

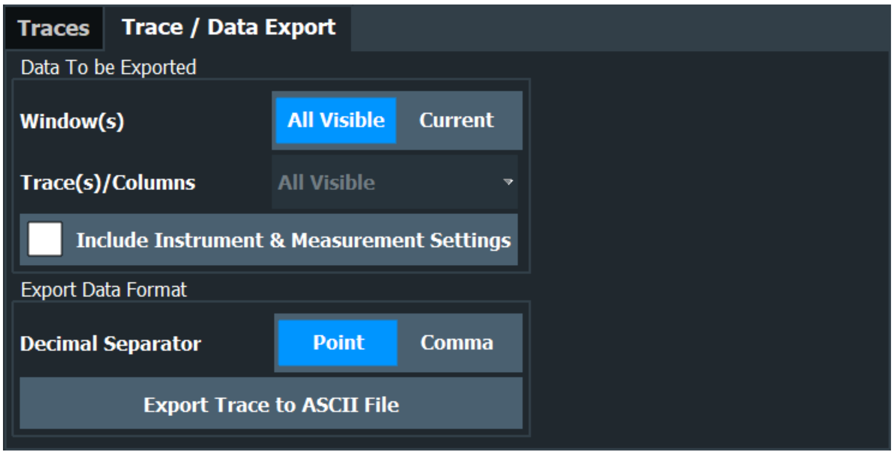
Predefined Trace Settings - Quick Config
Commonly required trace settings have been predefined and can be applied very quickly by selecting the appropriate button.

Function	Trace Settings	
Preset All Traces	Trace 1:	Clear Write
		Blank
Set Trace Mode Max | Avg | Min	Trace 1:	Max Hold
	Trace 2:	Average
	Trace 3:	Min Hold
		Blank
Set Trace Mode Max | ClrWrite | Min	Trace 1:	Max Hold
	Trace 2:	Clear Write
	Trace 3:	Min Hold
		Blank

5.1.2 Exporting Traces

Access: [TRACE] > "Trace Config" > "Trace / Data Export"

The functionality to export traces is similar to the Spectrum application. When you export a trace, the R&S FPS writes the trace data into an ASCII file. You can use the exported data for further evaluation in other programs like a spreadsheet.



The remote commands required to configure the trace export are described in Chapter 6.7.1, "Configuring Traces", on page 280.

Selecting data to export

The "Window(s)" toggle button selects the data that you want to export.

"All Visible" exports all traces in all result displays that are currently visible.

"Current" exports the traces in the currently selected (highlighted blue) result display.

If you export data from the currently selected result display, you can also select if you want to export all traces in that result display, or a single trace only from the "Trace(s) / Columns" dropdown menu.

Remote command:
`MMEMory:STORe<n>:TRACe` on page 282

Include Instrument & Measurement Settings
Includes additional instrument and measurement settings in the header of the export file for result data.

Remote command:
`FORMat:DEXPort:HEADer` on page 282

Decimal Separator
Defines the decimal separator for floating-point numerals for the data export/import files. Evaluation programs require different separators in different languages.

Remote command:
`FORMat:DEXPort:DSEParator` on page 281

Export Trace
The "Export Trace To ASCII File" button opens a dialog box to select a directory and file name for the ASCII file.

The results are output in the same order as they are displayed on the screen: window by window, trace by trace, and table row by table row.

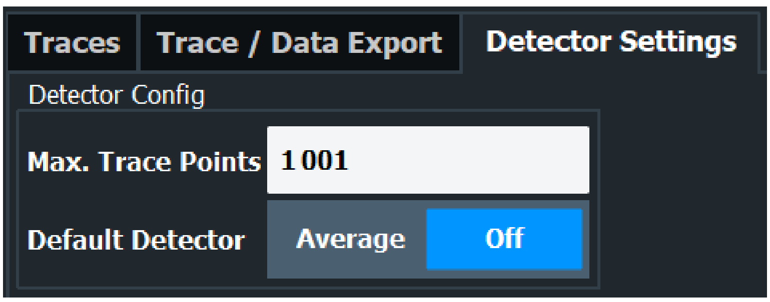
Note: Secure user mode.
In secure user mode, settings that are stored on the instrument are stored to volatile memory, which is restricted to 256 MB. Thus, a "memory limit reached" error can occur although the hard disk indicates that storage space is still available.

To store data permanently, select an external storage location such as a USB memory device.

For details, see "Protecting Data Using the Secure User Mode" in the "Data Management" section of the R&S FPS base unit user manual.

Remote command:
`MMEMory:STORe<n>:TRACe` on page 282

5.1.3 Detector Settings



Max. Trace Points
Sets the maximum number of trace points to be used by detectors.

Remote command:
`[SENSe:]DETector<t>:TRACe[:POINt]` on page 283

Default Detector
Selects the default detector for R&S FPS-K18 result displays.

Remote command:
`[SENSe:]DETector<t>:DEFault[:FUNCtion]` on page 283

5.2 Using Markers

The amplifier application provides four markers in most result displays.

5.2.1 Configuring Markers

Access: "Overview" > "Result Config" > "Marker Settings"

The "Marker Settings" contain settings that apply to all markers or have a general effect on marker functionality.

Marker Table Display
Defines how the marker information is displayed.

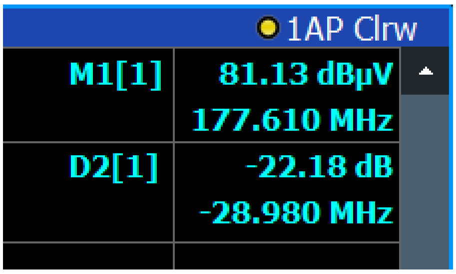
"On" Displays the marker information in a table in a separate area beneath the diagram.

"Off" No separate marker table is displayed.
If Marker Info is active, the marker information is displayed within the diagram area.

"Auto" (Default) If more than two markers are active, the marker table is displayed automatically.
If Marker Info is active, the marker information for up to two markers is displayed in the diagram area.

Remote command:
`DISPlay[:WINDow<n>]:MTABle` on page 286

Marker Info
Turns the marker information displayed in the diagram on and off.



Remote command:
`DISPlay[:WINDow<n>]:MINFo[:STATe]` on page 286

Link Markers Across Windows
Turns marker coupling across result windows on and off.

When you link markers, moving a marker in one result display moves the marker to the same sample in another window. This is useful to compare results in result displays that have different information on their x- and y-axis (for example the AM/AM and AM/PM results).

Remote command:
`CALCulate<n>:MARKer<m>:LINK` on page 285

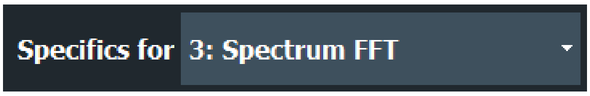
5.2.2 Configuring Individual Markers

Access: "Overview" > "Result Config" > "Markers"

The functionality to position markers and query their position is similar to the marker functionality available in the Spectrum application.

Availability of markers
The "Markers" and "Marker Settings" tabs are available for result displays that support markers.

If the tabs are unavailable, make sure to select a result display that actually supports markers from the "Specifics for:" dropdown menu (for example the spectrum FFT result display).



Note that the amplifier application does not support more than four markers in any result display.

Selected Marker

Marker name. The marker which is currently selected for editing is highlighted orange.

Remote command:
Marker selected via suffix <m> in remote commands.

Marker State

Activates or deactivates the marker in the diagram.

Remote command:
`CALCulate<n>:MARKer<m>[:STATe]` on page 291
`CALCulate<n>:DELTamarker<m>[:STATe]` on page 288

Marker Position X-value

Defines the position (x-value) of the marker in the diagram. For normal markers, the absolute position is indicated. For delta markers, the position relative to the reference marker is provided.

Remote command:
`CALCulate<n>:MARKer<m>:X` on page 291
`CALCulate<n>:DELTamarker<m>:X` on page 289

Marker Type

Toggles the marker type.

The type for marker 1 is always "Normal" , the type for delta marker 1 is always "Delta" . These types cannot be changed.

Note: If normal marker 1 is the active marker, switching the "Mkr Type" activates an additional delta marker 1. For any other marker, switching the marker type does not activate an additional marker, it only switches the type of the selected marker.

"Normal"	A normal marker indicates the absolute value at the defined position in the diagram.
"Delta"	A delta marker defines the value of the marker relative to the specified reference marker (marker 1 by default).

Remote command:
`CALCulate<n>:MARKer<m>[:STATe]` on page 291
`CALCulate<n>:DELTamarker<m>[:STATe]` on page 288

Reference Marker

Defines a marker as the reference marker which is used to determine relative analysis results (delta marker values).

Remote command:
`CALCulate<n>:DELTamarker<m>:MREFerence` on page 288

Linking to Another Marker
Links the current marker to the marker selected from the list of active markers. If the x-axis value of the initial marker is changed, the linked marker follows to the same position on the x-axis. Linking is off by default.

Using this function you can set two markers on different traces to measure the difference (e.g. between a max hold trace and a min hold trace or between a measurement and a reference trace).

Remote command:
`CALCulate<n>:MARKer<ms>:LINK:TO:MARKer<md>` on page 290
`CALCulate<n>:DELTamarker<ms>:LINK:TO:MARKer<md>` on page 288
`CALCulate<n>:DELTamarker<m>:LINK` on page 287

Assigning the Marker to a Trace
The "Trace" setting assigns the selected marker to an active trace. The trace determines which value the marker shows at the marker position. If the marker was previously assigned to a different trace, the marker remains on the previous frequency or time, but indicates the value of the new trace.

If a trace is turned off, the assigned markers and marker functions are also deactivated.

Remote command:
`CALCulate<n>:MARKer<m>:TRACe` on page 291

All Markers Off
Deactivates all markers in one step.

Remote command:
`CALCulate<n>:MARKer<m>:AOFF` on page 290

Marker Table Display
Defines how the marker information is displayed.

"On"	Displays the marker information in a table in a separate area beneath the diagram.
"Off"	No separate marker table is displayed. If Marker Info is active, the marker information is displayed within the diagram area.
"Auto"	(Default) If more than two markers are active, the marker table is displayed automatically. If Marker Info is active, the marker information for up to two markers is displayed in the diagram area.

Remote command:
`DISPlay[:WINDow<n>]:MTABle` on page 286

5.2.3 Positioning Markers

Peak Search

Sets the selected marker/delta marker to the maximum of the trace. If no marker is active, marker 1 is activated.

Remote command:
`CALCulate<n>:MARKer<m>:MAXimum[:PEAK]` on page 295
`CALCulate<n>:DELTamarker<m>:MAXimum[:PEAK]` on page 293

Search Next Peak

Sets the selected marker/delta marker to the next (lower) maximum of the assigned trace. If no marker is active, marker 1 is activated.

Remote command:
`CALCulate<n>:MARKer<m>:MAXimum:NEXT` on page 295
`CALCulate<n>:MARKer<m>:MAXimum:RIGHt` on page 295
`CALCulate<n>:MARKer<m>:MAXimum:LEFT` on page 294
`CALCulate<n>:DELTamarker<m>:MAXimum:NEXT` on page 293
`CALCulate<n>:DELTamarker<m>:MAXimum:RIGHt` on page 293
`CALCulate<n>:DELTamarker<m>:MAXimum:LEFT` on page 292

Search Minimum

Sets the selected marker/delta marker to the minimum of the trace. If no marker is active, marker 1 is activated.

Remote command:
`CALCulate<n>:MARKer<m>:MINimum[:PEAK]` on page 296
`CALCulate<n>:DELTamarker<m>:MINimum[:PEAK]` on page 294

Search Next Minimum

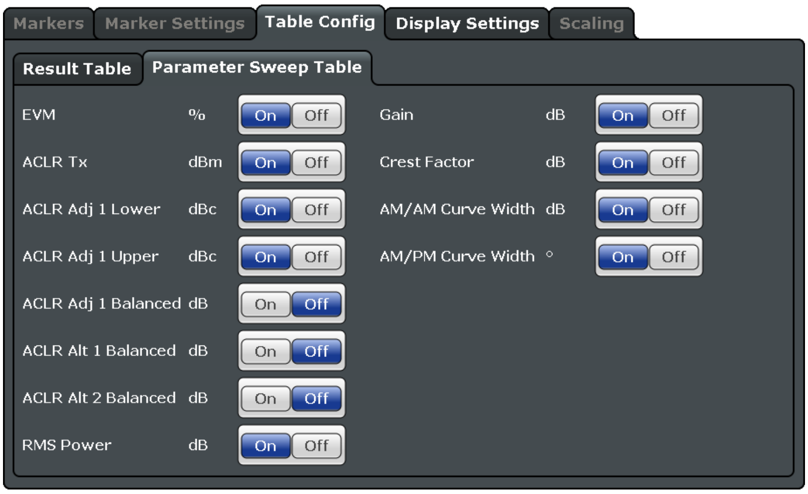
Sets the selected marker/delta marker to the next (higher) minimum of the selected trace. If no marker is active, marker 1 is activated.

Remote command:
`CALCulate<n>:MARKer<m>:MINimum:NEXT` on page 296
`CALCulate<n>:MARKer<m>:MINimum:LEFT` on page 295
`CALCulate<n>:MARKer<m>:MINimum:RIGHt` on page 296
`CALCulate<n>:DELTamarker<m>:MINimum:NEXT` on page 294
`CALCulate<n>:DELTamarker<m>:MINimum:LEFT` on page 293
`CALCulate<n>:DELTamarker<m>:MINimum:RIGHt` on page 294

5.3 Customizing Numerical Result Tables

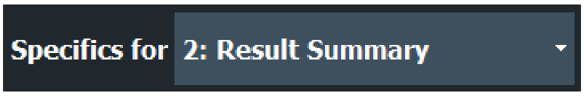
Access: "Overview" > "Result Config" > "Table Config"

By default, the application shows all supported numerical results in the result tables (result summary and parameter sweep table). However, you can add or remove results as you like.



Accessing the "Table Config" tab

Note that the "Table Config" tab is only available after you have selected the "Specifics for: Result Summary" or "Specifics for: Parameter Sweep Table" item from the corresponding dropdown menu at the bottom of the dialog box.



The dialog box for the result summary is made up out of three tabs:

- One for modulation accuracy results.
- One for power-related results.

The supported results of the parameter sweep table are part of a separate dialog box.

You can add or remove individual results by turning them "On" or "Off".

Remote command:

Individual result summary items: `DISPlay[:WINDow<n>]:TABLe:ITEM` on page 297

Individual parameter sweep items: `DISPlay[:WINDow<n>]:PTABle:ITEM` on page 296

All modulation accuracy items: `DISPlay[:WINDow<n>]:TABLe:ITEM:MACCuracy:ALL` on page 298

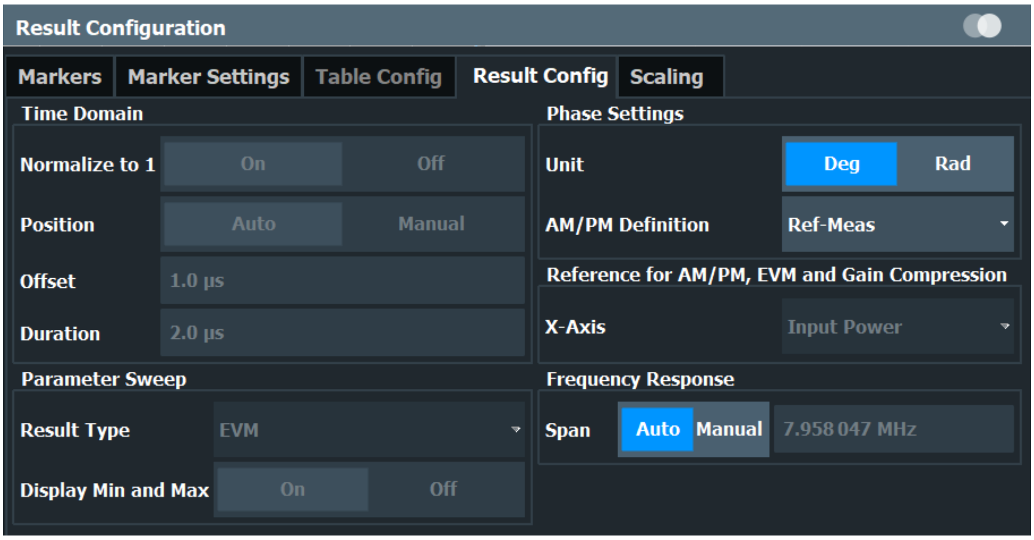
All power items: `DISPlay[:WINDow<n>]:TABLe:ITEM:POWer:ALL` on page 299

All parameter sweep items: `DISPlay[:WINDow<n>]:PTABle:ITEM:ALL` on page 297

5.4 Configuring Result Display Characteristics

Access: "Overview" > "Result Config" > "Display Settings"

The application allows you to define the information displayed in various graphical result displays.



Scope of the scaling

The functionality of the "Display Settings" is only available when you have selected one of the results displays that support this feature from the "Specifics for:" dropdown menu at the bottom of the dialog box.

(In this case, the functionality to adjust the time domain result display.)

Configuring the time domain result display

The "Time Domain" settings select the information displayed in the time domain result display and can thus be used to customize the diagram scale.

You can define the characteristics of the x-axis (the amount of displayed data) and the characteristics of the y-axis (normalized data or actual units).

Available when you select the time domain result display.

For more information, see "Time Domain" on page 29.

Remote command:
Normalization: `DISPlay[:WINDow<n>]:TDOMain:Y[:SCALe]:NORMalise[:STATe]` on page 304
Position: `DISPlay[:WINDow<n>]:TDOMain:X[:SCALe]:MODE` on page 303
Origin: `DISPlay[:WINDow<n>]:TDOMain:X[:SCALe]:OFFSet` on page 303
Duration: `DISPlay[:WINDow<n>]:TDOMain:X[:SCALe]:DURation` on page 302

Selecting the unit for phase results
You can display phase information either in degrees or radians. Select the preferred unit from the "Unit" dropdown menu.

The selected unit applies to the AM/PM results and the phase deviation vs time result.

Remote command:
`CALCulate<n>:UNIT:ANGLe` on page 301

Selecting the AM/PM definition
The "AM/PM Definition" feature selects the way the AM/PM results are calculated.

In effect, the "Meas - Ref" method is the inverse of the "Ref - Meas" method, which is the default.

Remote command:
`CALCulate<n>:AMPM:DEFinition` on page 300

Selecting the Power Reference
For several result displays, you can select the information that is referenced on the x-axis.

- AM/PM
- EVM vs Power
- Gain Compression

You can analyze these results either at the DUT input or at the DUT output. By default, the results show the information against the "Input Power".

To analyze the signal against the output power, select the "Output Power" item from the "X-Axis" dropdown menu.

Available when you select one of the result displays mentioned above.

Remote command:
`CALCulate<n>:PREFerence:X` on page 301

Selecting the result type displayed in the parameter sweep diagram
You can select one of several result types evaluated in the parameter sweep diagram. When you open more than one instance of the parameter sweep, you can select a different result for each of the instances.

For an extensive list of the supported result types, see "Parameter Sweep: Table" on page 19.

By default, the application shows the highest and lowest values that have been measured inside the diagram area.

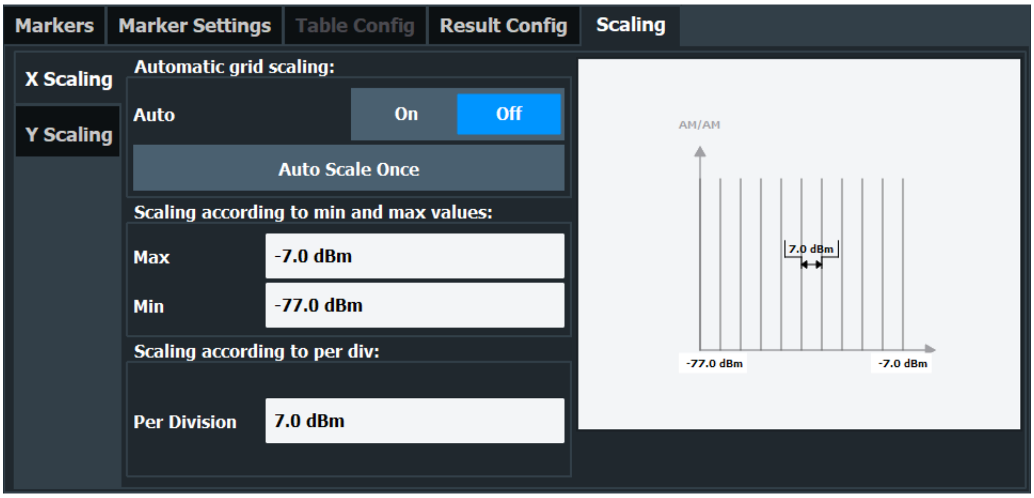
You can turn that off with the "Display Min and Max" feature.

Remote command:
`CONFigure:PSWeep:Z<n>:RESult` on page 302

5.5 Scaling the X-Axis

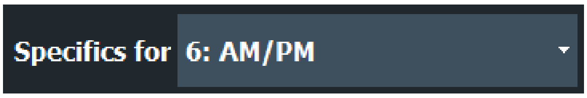
Access: "Overview" > "Result Config" > "Scaling" > "X Scaling"

By default, the application automatically scales the x-axis based on the current results. The scale changes when new measurement results are available. When you change the scale manually, the changes are shown in the diagram next to the settings.



Scope of the scaling

Scaling is applied only to the result display that you have selected from the "Specifics for:" dropdown menu at the bottom of the dialog box.



(In this case, the scale is applied to the AM/PM result display.)

Scaling the x-axis in particular is available for result displays that plot any kind of level values on both axes (for example the AM/PM result display).

Scaling the x-axis automatically

By default, the application scales the x-axis in all diagrams automatically (→ "Auto" = ON).

Automatic scaling tries to obtain the ideal scale for the current measurement results. The application adjusts the scale each time the results change.

You can also force an automatic scaling of the x-axis at any time with the "Auto Scale Once" function. When you select this function, the application scales the x-axis even if the results have not been changed.

Remote command:
`DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:X[:SCALe]:AUTO` on page 304

Scaling the x-axis manually
Settings for manual scaling of the x-axis become available when you turn automatic scaling off.

The application provides two methods to scale the x-axis.

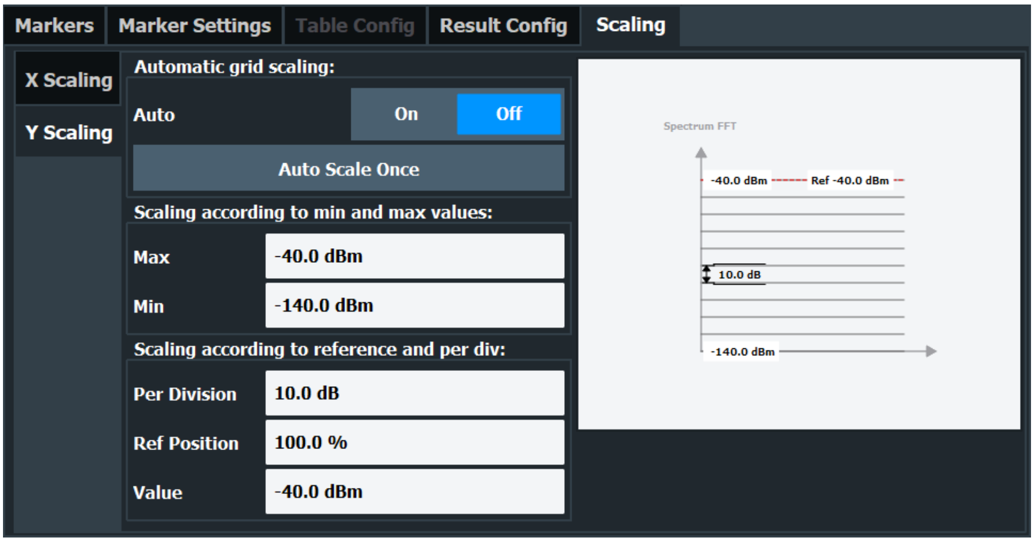
- Scaling according to minimum and maximum values
  The scale is defined by the values at the lower and upper end of the x-axis.
- Scaling according to the distance between two grid lines
  The scale is defined by the value range within two grid lines in the diagram (➙ per division). The distance between grid lines refers to diagrams that are split into 10 divisions.

Remote command:
Minimum: `DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:X[:SCALe]:MINimum` on page 305
Maximum: `DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:X[:SCALe]:MAXimum` on page 305
Distance: `DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:X[:SCALe]:PDIVision` on page 306

5.6 Scaling the Y-Axis

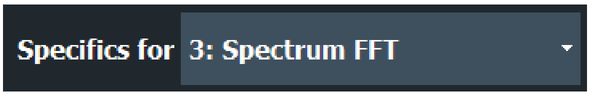
Access: "Overview" > "Result Config" > "Scaling" > "Y Scaling"

By default, the application automatically scales the y-axis based on the current results. The scale changes when new measurement results are available. When you change the scale manually, the changes are shown in the diagram next to the settings.



Scope of the scaling

Scaling is applied only to the result display that you have selected from the "Specifics for:" dropdown menu at the bottom of the dialog box.



(In this case, the scale is applied to the spectrum FFT result display.)

Scaling the y-axis automatically

By default, the application scales the y-axis in all diagrams automatically (→ "Auto" = ON).

Automatic scaling tries to obtain the ideal scale for the current measurement results. The application adjusts the scale each time the results change.

You can also force an automatic scaling of the y-axis at any time with the "Auto Scale Once" function. When you select this function, the application scales the y-axis even if the results have not been changed.

Remote command:
`DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:AUTO`
on page 307

Scaling the y-axis manually

Settings for manual scaling of the y-axis become available when you turn automatic scaling off.

The application provides two methods to scale the y-axis.

- Scaling according to minimum and maximum values
  The scale is defined by the values at the lower and upper end of the y-axis.
- Scaling according to reference value

The scale is defined relative to the reference value and a constant distance between the grid lines (→ per division). The distance between grid lines refers to diagrams that are split into 10 divisions.
The position of the reference value is arbitrary. By default it is at the upper end of the y-axis (100 %).

Remote command:
Minimum: `DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:MAXimum` on page 307
Maximum: `DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:MINimum` on page 307
Reference value: `DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:RVALue` on page 309
Position: `DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:RPOSition` on page 308
Distance: `DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:PDIVision` on page 308

6 Remote Control Commands for Amplifier Measurements

The following remote control commands are required to configure and perform amplifier measurements in a remote environment. The R&S FPS must already be set up for remote operation in a network as described in the base unit manual.

Universal functionality

Note that basic tasks that are also performed in the base unit in the same way are not described here. For a description of such tasks, see the R&S FPS User Manual.

In particular, this includes:

- Managing Settings and Results, i.e. storing and loading settings and result data.
- Basic instrument configuration, e.g. checking the system configuration, customizing the screen layout, or configuring networks and remote operation.
- Using the common status registers (specific status registers for Pulse measurements are not used).

6.1 Introduction

Commands are program messages that a controller (e.g. a PC) sends to the instrument or software. They operate its functions ('setting commands' or 'events') and request information ('query commands'). Some commands can only be used in one way, others work in two ways (setting and query). If not indicated otherwise, the commands can be used for settings and queries.

The syntax of a SCPI command consists of a header and, in most cases, one or more parameters. To use a command as a query, you have to append a question mark after the last header element, even if the command contains a parameter.

A header contains one or more keywords, separated by a colon. Header and parameters are separated by a "white space" (ASCII code 0 to 9, 11 to 32 decimal, e.g. blank). If there is more than one parameter for a command, these are separated by a comma from one another.

Only the most important characteristics that you need to know when working with SCPI commands are described here. For a more complete description, refer to the User Manual of the R&S FPS.

Remote command examples

Note that some remote command examples mentioned in this general introduction may not be supported by this particular application.

6.1.1 Conventions used in Descriptions

Note the following conventions 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 FPS 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.
- **Manual operation**
  If the result of a remote command can also be achieved in manual operation, a link to the description is inserted.

6.1.2 Long and Short Form

The keywords have a long and a short form. You can use either the long or the short form, but no other abbreviations of the keywords.

The short form is emphasized in upper case letters. Note however, that this emphasis only serves the purpose to distinguish the short from the long form in the manual. For the instrument, the case does not matter.

Example:

`SENSe:FREQuency:CENTer` is the same as `SENS:FREQ:CENT`.

6.1.3 Numeric Suffixes

Some keywords have a numeric suffix if the command can be applied to multiple instances of an object. In that case, the suffix selects a particular instance (e.g. a measurement window).

Numeric suffixes are indicated by angular brackets (<n>) next to the keyword.

If you don't quote a suffix for keywords that support one, a 1 is assumed.

Example:

`DISPlay[:WINDow<1...4>]:ZOOM:STATe` enables the zoom in a particular measurement window, selected by the suffix at `WINDow`.

`DISPlay:WINDow4:ZOOM:STATe ON` refers to window 4.

6.1.4 Optional Keywords

Some keywords are optional and are only part of the syntax because of SCPI compliance. You can include them in the header or not.

Note that if an optional keyword has a numeric suffix and you need to use the suffix, you have to include the optional keyword. Otherwise, the suffix of the missing keyword is assumed to be the value 1.

Optional keywords are emphasized with square brackets.

Example:

Without a numeric suffix in the optional keyword:

`[SENSe:]FREQuency:CENTer` is the same as `FREQuency:CENTer`

With a numeric suffix in the optional keyword:

`DISPlay[:WINDow<1...4>]:ZOOM:STATe`

`DISPlay:ZOOM:STATe ON` enables the zoom in window 1 (no suffix).

`DISPlay:WINDow4:ZOOM:STATe ON` enables the zoom in window 4.

6.1.5 Alternative Keywords

A vertical stroke indicates alternatives for a specific keyword. You can use both keywords to the same effect.

Example:

`[SENSe:]BANDwidth|BWIDth[:RESolution]`

In the short form without optional keywords, `BAND 1MHZ` would have the same effect as `BWID 1MHZ`.

6.1.6 SCPI Parameters

Many commands feature one or more parameters.

If a command supports more than one parameter, these are separated by a comma.

Example:

`LAYout:ADD:WINDow Spectrum,LEFT,MTABle`

Parameters may have different forms of values.

6.1.6.1 Numeric Values

Numeric values can be entered in any form, i.e. with sign, decimal point or exponent. In case of physical quantities, you can also add the unit. If the unit is missing, the command uses the basic unit.

Example:

With unit: `SENSe:FREQuency:CENTer 1GHZ`

Without unit: `SENSe:FREQuency:CENTer 1E9` would also set a frequency of 1 GHz.

Values exceeding the resolution of the instrument are rounded up or down.

If the number you have entered is not supported (e.g. in case of discrete steps), the command returns an error.

Instead of a number, you can also set numeric values with a text parameter in special cases.

- MIN/MAX
  Defines the minimum or maximum numeric value that is supported.
- DEF
  Defines the default value.
- UP/DOWN
  Increases or decreases the numeric value by one step. The step size depends on the setting. In some cases you can customize the step size with a corresponding command.

Querying numeric values

When you query numeric values, the system returns a number. In case of physical quantities, it applies the basic unit (e.g. Hz in case of frequencies). The number of digits after the decimal point depends on the type of numeric value.

Example:

Setting: `SENSe:FREQuency:CENTer 1GHZ`

Query: `SENSe:FREQuency:CENTer?` would return `1E9`

In some cases, numeric values may be returned as text.

- INF/NINF
  Infinity or negative infinity. Represents the numeric values 9.9E37 or -9.9E37.
- NAN
  Not a number. Represents the numeric value 9.91E37. NAN is returned in case of errors.

6.1.6.2 Boolean

Boolean parameters represent two states. The "ON" state (logically true) is represented by "ON" or a numeric value 1. The "OFF" state (logically untrue) is represented by "OFF" or the numeric value 0.

Querying Boolean parameters

When you query Boolean parameters, the system returns either the value 1 ("ON") or the value 0 ("OFF").

Example:

Setting: `DISPlay:WINDow:ZOOM:STATe ON`

Query: `DISPlay:WINDow:ZOOM:STATe?` would return `1`

6.1.6.3 Character Data

Character data follows the syntactic rules of keywords. You can enter text using a short or a long form. For more information see Chapter 6.1.2, "Long and Short Form", on page 109.

Querying text parameters

When you query text parameters, the system returns its short form.

Example:

Setting: `SENSe:BANDwidth:RESolution:TYPE NORMal`

Query: `SENSe:BANDwidth:RESolution:TYPE?` would return `NORM`

6.1.6.4 Character Strings

Strings are alphanumeric characters. They have to be in straight quotation marks. You can use a single quotation mark (') or a double quotation mark (").

Example:

`INSTRument:DELete 'Spectrum'`

6.1.6.5 Block Data

Block data is a format which is suitable for the transmission of large amounts of data.

The ASCII character # introduces the data block. The next number indicates how many of the following digits describe the length of the data block. In the example the 4 following digits indicate the length to be 5168 bytes. The data bytes follow. During the transmission of these data bytes all end or other control signs are ignored until all bytes are transmitted. #0 specifies a data block of indefinite length. The use of the indefinite format requires an `NL^END` message to terminate the data block. This format is useful when the length of the transmission is not known or if speed or other considerations prevent segmentation of the data into blocks of definite length.

6.2 Common Suffixes

In the Amplifier measurement application, the following common suffixes are used in remote commands:

Table 6-1: Common suffixes used in remote commands in the Amplifier measurement application

Suffix	Value range	Description
<m>	1..16	Marker
<n>	1..16	Window (**in the currently selected channel**)
<t>	1..6	Trace
	1 to 8	Limit line

6.3 Selecting the Application

INSTrument:CREate:DUPLicate

This command duplicates the currently selected channel, i.e creates a new channel of the same type and with the identical measurement settings. The name of the new channel is the same as the copied channel, extended by a consecutive number (e.g. "IQAnalyzer" -> "IQAnalyzer 2").

The channel to be duplicated must be selected first using the `INST:SEL` command.

Example:
`INST:SEL 'IQAnalyzer'`
`INST:CRE:DUPL`
Duplicates the channel named 'IQAnalyzer' and creates a new channel named 'IQAnalyzer2'.

Usage: Event

INSTrument:CREate:REPLace <ChannelName1>,<ChannelType>,<ChannelName2>

This command replaces a channel with another one.

Setting parameters:

<ChannelName1> String containing the name of the channel you want to replace.

<ChannelType> Channel type of the new channel.
For a list of available channel types see `INSTrument:LIST?` on page 115.

<ChannelName2> String containing the name of the new channel.
Note: If the specified name for a new channel already exists, the default name, extended by a sequential number, is used for the new channel (see `INSTrument:LIST?` on page 115).
Channel names can have a maximum of 31 characters, and must be compatible with the Windows conventions for file names. In particular, they must not contain special characters such as ":", "*", "?".

Example:
`INST:CRE:REPL 'IQAnalyzer2',IQ,'IQAnalyzer'`
Replaces the channel named "IQAnalyzer2" by a new channel of type "IQ Analyzer" named "IQAnalyzer".

Usage: Setting only

INSTrument:CREate[:NEW] <ChannelType>, <ChannelName>

This command adds an additional measurement channel. You can configure up to 10 measurement channels at the same time (depending on available memory).

Parameters:

<ChannelType> Channel type of the new channel.
For a list of available channel types see `INSTrument:LIST?` on page 115.

<ChannelName> String containing the name of the channel.
Note that you can not assign an existing channel name to a new channel; this will cause an error.

Example: `INST:CRE SAN, 'Spectrum 2'`
Adds an additional spectrum display named "Spectrum 2".

INSTrument:DELete <ChannelName>

This command deletes a channel.

If you delete the last channel, the default "Spectrum" channel is activated.

Setting parameters:
<ChannelName> String containing the name of the channel you want to delete.
A channel must exist in order to be able delete it.

Example: `INST:DEL 'IQAnalyzer4'`
Deletes the channel with the name 'IQAnalyzer4'.

Usage: Setting only

INSTrument:LIST?

This command queries all active channels. This is useful in order to obtain the names of the existing channels, which are required in order to replace or delete the channels.

Return values:
<ChannelType>, <ChannelName> For each channel, the command returns the channel type and channel name (see tables below).
Tip: to change the channel name, use the `INSTrument:REName` command.

Example: `INST:LIST?`
Result for 3 channels:
`'ADEM','Analog Demod','IQ','IQ Analyzer','IQ','IQ Analyzer2'`

Usage: Query only

Table 6-2: Available channel types and default channel names in Signal and Spectrum Analyzer mode

Application	<ChannelType> parameter	Default Channel name*)
Spectrum	SANALYZER	Spectrum
1xEV-DO BTS (R&S FPS-K84)	BDO	1xEV-DO BTS
1xEV-DO MS (R&S FPS-K85)	MDO	1xEV-DO MS
3GPP FDD BTS (R&S FPS-K72)	BWCD	3G FDD BTS
3GPP FDD UE (R&S FPS-K73)	MWCD	3G FDD UE
*) If the specified name for a new channel already exists, the default name, extended by a sequential number, is used for the new channel.		

Application	<ChannelType> parameter	Default Channel name*)
Analog Modulation Analysis (R&S FPS-K7)	ADEM	Analog Demod
cdma2000 BTS (R&S FPS-K82)	BC2K	CDMA2000 BTS
cdma2000 MS (R&S FPS-K83)	MC2K	CDMA2000 MS
GSM (R&S FPS-K10)	GSM	GSM
I/Q Analyzer	IQ	IQ Analyzer
LTE (R&S FPS-K10x)	LTE	LTE
NB-IoT (R&S FPS-K106)	NIOT	NB-IoT
Noise (R&S FPS-K30)	NOISE	Noise
5G NR (R&S FPS-K144)	NR5G	5G NR
Phase Noise (R&S FPS-K40)	PNOISE	Phase Noise
TD-SCDMA BTS (R&S FPS-K76)	BTDS	TD-SCDMA BTS
TD-SCDMA UE (R&S FPS-K77)	MTDS	TD-SCDMA UE
Verizon 5GTF Measurement Application (V5GTF, R&S FPS-K118)	V5GT	V5GT
VSA (R&S FPS-K70)	DDEM	VSA
WLAN (R&S FPS-K91)	WLAN	WLAN
*) If the specified name for a new channel already exists, the default name, extended by a sequential number, is used for the new channel.		

INSTrument:REName <ChannelName1>, <ChannelName2>

This command renames a channel.

Setting parameters:

<ChannelName1> String containing the name of the channel you want to rename.

<ChannelName2> String containing the new channel name.
Note that you cannot assign an existing channel name to a new channel; this will cause an error.
Channel names can have a maximum of 31 characters, and must be compatible with the Windows conventions for file names. In particular, they must not contain special characters such as ":", "*", "?".

Example:
```
INST:REN 'IQAnalyzer2','IQAnalyzer3'
```
Renames the channel with the name 'IQAnalyzer2' to 'IQAnalyzer3'.

Usage: Setting only

INSTrument[:SELect] <ChannelType> | <ChannelName>

This command activates a new channel with the defined channel type, or selects an existing channel with the specified name.

Also see

- `INSTrument:CREate[:NEW]` on page 114

Parameters:

<ChannelType> Channel type of the new channel.
For a list of available channel types see `INSTrument:LIST?` on page 115.

<ChannelName> String containing the name of the channel.

Example: `INST IQ`
Activates a channel for the I/Q Analyzer application (evaluation mode).
`INST 'MyIQSpectrum'`
Selects the channel named 'MyIQSpectrum' (for example before executing further commands for that channel).

SYSTem:PRESet:CHANnel[:EXEC]

This command restores the default instrument settings in the current channel.

Use `INST:SEL` to select the channel.

Example: `INST:SEL 'Spectrum2'`
Selects the channel for "Spectrum2".
`SYST:PRES:CHAN:EXEC`
Restores the factory default settings to the "Spectrum2"channel.

Usage: Event

Manual operation: See "Preset Channel" on page 33

6.4 Configuring the Screen Layout

DISPlay:FORMat <Format>

This command determines which tab is displayed.

Parameters:

<Format> **SPLit**
Displays the MultiView tab with an overview of all active channels

SINGle
Displays the measurement channel that was previously focused.

*RST: SING

Example: `DISP:FORM SPL`

DISPlay[:WINDow<n>]:SIZE <Size>

This command maximizes the size of the selected result display window *temporarily*. To change the size of several windows on the screen permanently, use the `LAY:SPL` command (see `LAYout:SPLitter` on page 122).

Suffix:

<n> Window

Parameters:

<Size> **LARGe**
Maximizes the selected window to full screen.
Other windows are still active in the background.

SMALl
Reduces the size of the selected window to its original size.
If more than one measurement window was displayed originally, these are visible again.

*RST: SMALl

Example: `DISP:WIND2:SIZE LARG`

LAYout:ADD[:WINDow]? <WindowName>,<Direction>,<WindowType>

This command adds a window to the display in the active channel.

This command is always used as a query so that you immediately obtain the name of the new window as a result.

To replace an existing window, use the `LAYout:REPLace[:WINDow]` command.

Query parameters:

<WindowName> String containing the name of the existing window the new window is inserted next to.
By default, the name of a window is the same as its index. To determine the name and index of all active windows, use the `LAYout:CATalog[:WINDow]?` query.

<Direction> LEFT | RIGHt | ABOVe | BELow
Direction the new window is added relative to the existing window.

<WindowType> text value
Type of result display (evaluation method) you want to add.
See the table below for available parameter values.

Return values:

<NewWindowName> When adding a new window, the command returns its name (by default the same as its number) as a result.

Example: `LAY:ADD? '1',LEFT,MTAB`
Result:
`'2'`
Adds a new window named '2' with a marker table to the left of window 1.

Usage: Query only

Manual operation: See "Adjacent Channel Leakage Error (ACLR)" on page 11
See "AM/AM" on page 12
See "AM/PM" on page 13
See "EVM vs Power" on page 14
See "Error Vector Spectrum" on page 15
See "Gain Compression" on page 16
See "Gain Deviation vs Time" on page 17
See "Magnitude Capture" on page 17
See "Phase Deviation vs Time" on page 20
See "Raw EVM" on page 20
See "Numeric Result Summary" on page 21
See "Spectrum FFT" on page 28
See "Time Domain" on page 29
See "Statistics Table" on page 30

Table 6-3: <WindowType> parameter values for Amplifier Measurement application

Parameter value	Window type
ACP	Adjacent Channel Power (Table)
AMAM	AM/AM
AMPM	AM/PM
GCOMpression	Gain Compression
GDVT	Gain Deviation vs Time
MTABle	Marker Table

Parameter value	Window type
PSWeep	Parameter Sweep (Diagram)
PTABle	Parameter Sweep (Table)
PDVT	Phase Deviation vs Time
REVM	Raw EVM
RFMagnitude	Magnitude Capture RF
RFSPectrum	Spectrum FFT
RTABle	Result Summary (Table)
SEVM	Spectrum EVM
TDOMain	Time Domain

LAYout:CATalog[:WINDow]?

This command queries the name and index of all active windows in the active channel from top left to bottom right. The result is a comma-separated list of values for each window, with the syntax:

<WindowName_1>,<WindowIndex_1>..<WindowName_n>,<WindowIndex_n>

Return values:

<WindowName> string

Name of the window.
In the default state, the name of the window is its index.

<WindowIndex> **numeric value**
Index of the window.

Example: `LAY:CAT?`
Result:
`'2',2,'1',1`
Two windows are displayed, named '2' (at the top or left), and '1' (at the bottom or right).

Usage: Query only

LAYout:DIRection <Direction>

This command selects the general direction of the smart grid.

Parameters:

<Direction> **HORizontal**

VERTical

*RST: HORizontal

Example: `LAY:DIR HOR`

LAYout:IDENtify[:WINDow]? <WindowName>

This command queries the **index** of a particular display window in the active channel.

Note: to query the **name** of a particular window, use the `LAYout:WINDow<n>:IDENtify?` query.

Query parameters:

<WindowName> String containing the name of a window.

Return values:

<WindowIndex> Index number of the window.

Example: `LAY:WIND:IDEN? '2'`
Queries the index of the result display named '2'.
Response:
`2`

Usage: Query only

LAYout:REMove[:WINDow] <WindowName>

This command removes a window from the display in the active channel.

Setting parameters:

<WindowName> String containing the name of the window. In the default state, the name of the window is its index.

Example: `LAY:REM '2'`
Removes the result display in the window named '2'.

Usage: Setting only

LAYout:REPLace[:WINDow] <WindowName>,<WindowType>

This command replaces the window type (for example from "Diagram" to "Result Summary") of an already existing window in the active channel while keeping its position, index and window name.

To add a new window, use the `LAYout:ADD[:WINDow]?` command.

Setting parameters:

<WindowName> String containing the name of the existing window.
By default, the name of a window is the same as its index. To determine the name and index of all active windows in the active channel, use the `LAYout:CATalog[:WINDow]?` query.

<WindowType> Type of result display you want to use in the existing window.
See `LAYout:ADD[:WINDow]?` on page 118 for a list of available window types.

Example: `LAY:REPL:WIND '1',MTAB`
Replaces the result display in window 1 with a marker table.

Usage: Setting only

LAYout:SPLitter <Index1>, <Index2>, <Position>

This command changes the position of a splitter and thus controls the size of the windows on each side of the splitter.

Compared to the `DISPlay[:WINDow<n>]:SIZE` on page 118 command, the `LAYout:SPLitter` changes the size of all windows to either side of the splitter permanently, it does not just maximize a single window temporarily.

Note that windows must have a certain minimum size. If the position you define conflicts with the minimum size of any of the affected windows, the command will not work, but does not return an error.

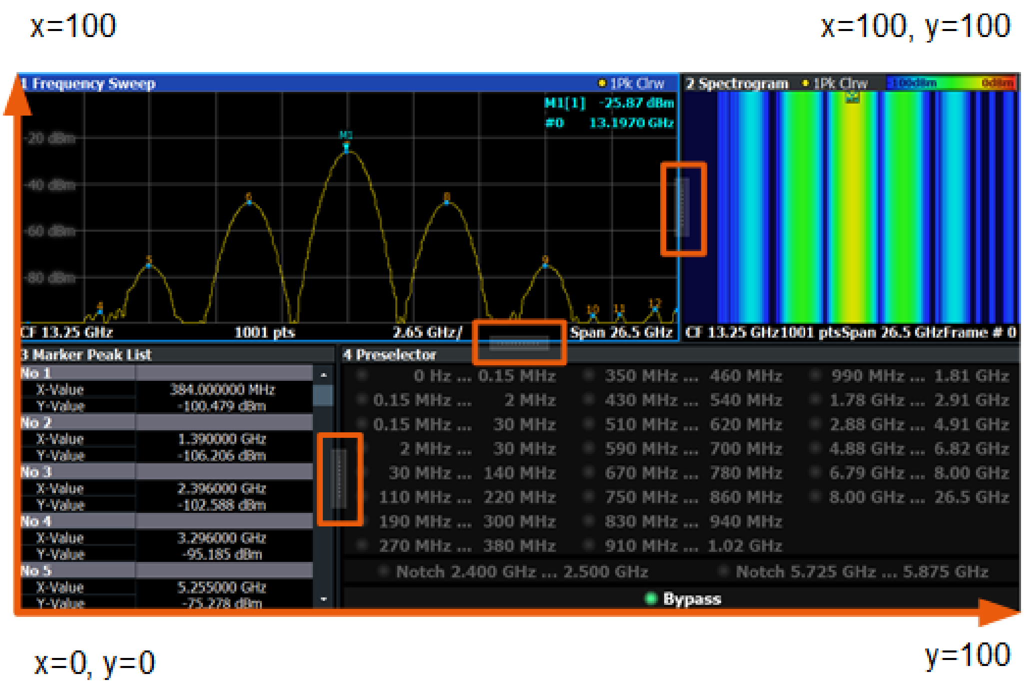


Figure 6-1: SmartGrid coordinates for remote control of the splitters

Setting parameters:

<Index1>	The index of one window the splitter controls.
<Index2>	The index of a window on the other side of the splitter.
<Position>	New vertical or horizontal position of the splitter as a fraction of the screen area (without channel and status bar and softkey menu). The point of origin (x = 0, y = 0) is in the lower left corner of the screen. The end point (x = 100, y = 100) is in the upper right corner of the screen. (See Figure 6-1.) The direction in which the splitter is moved depends on the screen layout. If the windows are positioned horizontally, the splitter also moves horizontally. If the windows are positioned vertically, the splitter also moves vertically. Range: 0 to 100

Example: `LAY:SPL 1,3,50`
Moves the splitter between window 1 ('Frequency Sweep') and 3 ('Marker Table') to the center (50%) of the screen, i.e. in the figure above, to the left.

Example: `LAY:SPL 1,4,70`
Moves the splitter between window 1 ('Frequency Sweep') and 3 ('Marker Peak List') towards the top (70%) of the screen.
The following commands have the exact same effect, as any combination of windows above and below the splitter moves the splitter vertically.
`LAY:SPL 3,2,70`
`LAY:SPL 4,1,70`
`LAY:SPL 2,1,70`

Usage: Setting only

LAYout:WINDow<n>:ADD? <Direction>,<WindowType>

This command adds a measurement window to the display. Note that with this command, the suffix <n> determines the existing window next to which the new window is added, as opposed to `LAYout:ADD[:WINDow]?`, for which the existing window is defined by a parameter.

To replace an existing window, use the `LAYout:WINDow<n>:REPLace` command.

This command is always used as a query so that you immediately obtain the name of the new window as a result.

Suffix:
<n> Window

Query parameters:
<Direction> LEFT | RIGHt | ABOVe | BELow

<WindowType> Type of measurement window you want to add.
See `LAYout:ADD[:WINDow]?` on page 118 for a list of available window types.

Return values:
<NewWindowName> When adding a new window, the command returns its name (by default the same as its number) as a result.

Example: `LAY:WIND1:ADD? LEFT,MTAB`
Result:
`'2'`
Adds a new window named '2' with a marker table to the left of window 1.

Usage: Query only

LAYout:WINDow<n>:IDENtify?

This command queries the **name** of a particular display window (indicated by the <n> suffix) in the active channel.

Note: to query the **index** of a particular window, use the `LAYout:IDENtify[:WINDow]?` command.

Suffix:

<n> Window

Return values:

<WindowName> String containing the name of a window.
In the default state, the name of the window is its index.

Example: `LAY:WIND2:IDEN?`
Queries the name of the result display in window 2.
Response:
`'2'`

Usage: Query only

LAYout:WINDow<n>:REMove

This command removes the window specified by the suffix <n> from the display in the active channel.

The result of this command is identical to the `LAYout:REMove[:WINDow]` command.

Suffix:

<n> Window

Example: `LAY:WIND2:REM`
Removes the result display in window 2.

Usage: Event

LAYout:WINDow<n>:REPLace <WindowType>

This command changes the window type of an existing window (specified by the suffix <n>) in the active channel.

The effect of this command is identical to the `LAYout:REPLace[:WINDow]` command.

To add a new window, use the `LAYout:WINDow<n>:ADD?` command.

Suffix:

<n> Window

Setting parameters:

<WindowType> Type of measurement window you want to replace another one with.
See `LAYout:ADD[:WINDow]?` on page 118 for a list of available window types.

Example: `LAY:WIND2:REPL MTAB`
Replaces the result display in window 2 with a marker table.

Usage: Setting only

LAYout:WINDow<n>:TYPE?

Queries the window type of the window specified by the index <n>. For a list of possible window types see `LAYout:ADD[:WINDow]?` on page 118.

Suffix:
<n> Window

Example: `LAY:WIND2:TYPE?`
Response:
`MACC`
Modulation accuracy

Usage: Query only

6.5 Performing Amplifier Measurements

6.5.1 Performing Measurements

You can include the Amplifier measurements in a sequence of measurements. For a comprehensive description of commands required to do so, refer to the R&S FPS User Manual.

INITiate<n>:CONMeas

This command restarts a (single) measurement that has been stopped (using `ABORt`) or finished in single measurement mode.

The measurement is restarted at the beginning, not where the previous measurement was stopped.

As opposed to INITiate<n>[:IMMediate], this command does not reset traces in maxhold, minhold or average mode. Therefore it can be used to continue measurements using maxhold or averaging functions.

Suffix:
<n> irrelevant

Manual operation: See " Continue Single Sweep " on page 34

INITiate<n>:CONTinuous <State>

This command controls the measurement mode for an individual channel.

Note that in single measurement mode, you can synchronize to the end of the measurement with *OPC, *OPC? or *WAI. In continuous measurement mode, synchronization to the end of the measurement is not possible. Thus, it is not recommended that you use continuous measurement mode in remote control, as results like trace data or markers are only valid after a single measurement end synchronization.

Suffix:
<n> irrelevant

Parameters:
<State> ON | OFF | 0 | 1

ON | 1
Continuous measurement

OFF | 0
Single measurement

*RST: 0

Example: `INIT:CONT OFF`
Switches the measurement mode to single measurement.
`INIT:CONT ON`
Switches the measurement mode to continuous measurement.

Manual operation: See " Continuous Sweep / Run Cont " on page 34

INITiate<n>[:IMMediate]

This command starts a (single) new measurement.

You can synchronize to the end of the measurement with *OPC, *OPC? or *WAI.

Suffix:
<n> irrelevant

Manual operation: See " Single Sweep / Run Single " on page 34

INITiate:SEQuencer:ABORt

This command stops the currently active sequence of measurements.

You can start a new sequence any time using INITiate:SEQuencer:IMMediate on page 127.

Usage: Event

INITiate:SEQuencer:IMMediate

This command starts a new sequence of measurements by the Sequencer.

Its effect is similar to the INITiate<n>[:IMMediate] command used for a single measurement.

Before this command can be executed, the Sequencer must be activated (see SYSTem:SEQuencer on page 128).

Example:

`SYST:SEQ ON`
Activates the Sequencer.
`INIT:SEQ:MODE SING`
Sets single sequence mode so each active measurement will be performed once.
`INIT:SEQ:IMM`
Starts the sequential measurements.

INITiate:SEQuencer:MODE <Mode>

Defines the capture mode for the entire measurement sequence and all measurement groups and channels it contains.

Note: In order to synchronize to the end of a measurement sequence using *OPC, *OPC? or *WAI you must use SINGle Sequence mode.

Parameters:

<Mode>

SINGle
Each measurement group is started one after the other in the order of definition. All measurement channels in a group are started simultaneously and performed once. After *all* measurements are completed, the next group is started. After the last group, the measurement sequence is finished.

CONTinuous
Each measurement group is started one after the other in the order of definition. All measurement channels in a group are started simultaneously and performed once. After *all* measurements are completed, the next group is started. After the last group, the measurement sequence restarts with the first one and continues until it is stopped explicitly.

*RST: CONTinuous

SYSTem:SEQuencer <State>

This command turns the Sequencer on and off. The Sequencer must be active before any other Sequencer commands (`INIT:SEQ...`) are executed, otherwise an error will occur.

Parameters:

<State> ON | OFF | 0 | 1

ON | 1
The Sequencer is activated and a sequential measurement is started immediately.

OFF | 0
The Sequencer is deactivated. Any running sequential measurements are stopped. Further Sequencer commands (`INIT:SEQ...`) are not available.

*RST: 0

Example:

`SYST:SEQ ON`
Activates the Sequencer.
`INIT:SEQ:MODE SING`
Sets single Sequencer mode so each active measurement will be performed once.
`INIT:SEQ:IMM`
Starts the sequential measurements.
`SYST:SEQ OFF`

6.5.2 Retrieving Graphical Measurement Results

FORMat[:DATA] <Format>[, <BitLength>]

This command selects the data format that is used for transmission of trace data from the R&S FPS to the controlling computer.

Note that the command has no effect for data that you send to the R&S FPS. The R&S FPS automatically recognizes the data it receives, regardless of the format.

Parameters:

<Format>

ASCii
ASCii format, separated by commas.
This format is almost always suitable, regardless of the actual data format. However, the data is not as compact as other formats may be.

REAL
Floating-point numbers (according to IEEE 754) in the "definite length block format".
In the Spectrum application, the format setting `REAL` is used for the binary transmission of trace data.

<BitLength> Length in bits for floating-point results

16
16-bit floating-point numbers.
Compared to `REAL,32` format, half as many numbers are returned.

32
32-bit floating-point numbers
For I/Q data, 8 bytes per sample are returned for this format setting.

64
64-bit floating-point numbers
Compared to `REAL,32` format, twice as many numbers are returned.

Example: `FORM REAL,32`

TRACe<n>[:DATA]? <Trace>

This command queries the measurement results in the graphical result displays. Usually, the measurement results are either displayed on the y-axis (two-dimensional diagrams) or the z-axis (three-dimensional diagrams).

Suffix:
<n> 1..n
Window

Query parameters:
<Trace> TRACE1 | ... | TRACE6

Selects the trace to be queried.
Note that the available number of traces depends on the result display.
For example, the Magnitude Capture result display only supports TRACE1, while the Time Domain result display supports TRACE1 to TRACE6.

Return values:
<Result> <numeric value>

Values of the captured samples in chronological order.

Example: `TRAC:DATA TRACE1`
Queries the results displayed on trace 1.

Usage: Query only

Manual operation: See "AM/AM" on page 12
See "AM/PM" on page 13
See "EVM vs Power" on page 14
See "Error Vector Spectrum" on page 15
See "Gain Compression" on page 16
See "Gain Deviation vs Time" on page 17
See "Magnitude Capture" on page 17
See "Phase Deviation vs Time" on page 20
See "Raw EVM" on page 20
See "Spectrum FFT" on page 28
See "Time Domain" on page 29

TRACe<n>[:DATA]:X? <Trace>

This command queries the measurement results as displayed on the x-axis in the graphical result displays.

Suffix:
<n> 1..n
Window

Query parameters:
<Trace> TRACE1 | ... | TRACE6

Selects the trace to be queried.
Note that the available number of traces depends on the result display.
For example, the Magnitude Capture result display only supports TRACE1, while the Time Domain result display supports TRACE1 to TRACE6.

Return values:
<Result> <numeric value>

X-axis values of the captured samples in chronological order.

Example: `TRAC:DATA TRACE1`
Queries the results displayed on trace 1.

Usage: Query only

TRACe<n>[:DATA]:Y? <Trace>

This command queries the measurement results as displayed on the y-axis in result displays with three axes (for example the Parameter Sweep).

Suffix:
<n> 1..n
Window

Query parameters:

<Trace> TRACE1 | ... | TRACE6

Selects the trace to be queried.
Note that the available number of traces depends on the result display.

Example: `TRAC:DATA TRACE1`
Queries the results displayed on trace 1.

Usage: Query only

6.5.3 Retrieving Numeric Results

6.5.3.1 Retrieving General Numeric Results

FETCh:TTS:CURRent[:RESult]?

This command queries the trigger to sync result.

This is the time from start of capture (i.e. including pre-trigger samples) to the start of the sync range, which is not necessarily the beginning of the reference waveform.

Return values:

<Time> <numeric value>

Default unit: s

Example: `FETCh:TTS:CURRent[:RESult]?`

Usage: Query only

6.5.3.2 Retrieving Results of the Result Summary

Retrieving All Results

FETCh:MACCuracy[:RESult]:ALL?

This command queries all numerical results shown in the Result Summary.

Return values:

<Results> <numerical value>: Results as a comma separated list.
The order of results is the same as in the result summary:
`<RawEVMMin>, <RawEVMCurrent>, <RawEVMMax>, <RawModelEVMMin>, <RawModelEVMCurrent>, <RawModelEVMMax>,...`
The unit depends on the result.
If a result hasn't been calculated, the command returns `NAN`.

Example: `FETC:MACC:ALL?`
would return, e.g.
`0.277,0.277,0.277,0.002,0.245,0.922,...`

Usage: Query only

FETCh:POWer[:RESult]:ALL?

This command queries all power related numerical results as shown in the result summary.

Return values:

<Results> <numerical value>: Results as a comma separated list.
The order of results is the same as in the result summary:
The unit depends on the result.
If a result hasn't been calculated, the command returns `NAN`.

Example: `FETC:POW:ALL?`
would return, e.g.

Usage: Query only

Retrieving the Modulation Accuracy

FETCh:MACCuracy:FERRor:MAXimum[:RESult]?
FETCh:MACCuracy:FERRor:MINimum[:RESult]?
FETCh:MACCuracy:FERRor:CURRent[:RESult]?

This command queries the Frequency Error as shown in the Result Summary.

Return values:

<FrequencyError> <numeric value>

Minimum, maximum or current Frequency Error, depending on the command syntax.

Default unit: Hz

Example: `FETC:MACC:FERR:MAX?`
would return, e.g.
`1.2879`

Usage: Query only

Manual operation: See "Results to check modulation accuracy" on page 23

FETCh:MACCuracy:GIMBalance:MAXimum[:RESult]?
FETCh:MACCuracy:GIMBalance:MINimum[:RESult]?
FETCh:MACCuracy:GIMBalance:CURRent[:RESult]?

This command queries the Gain Imbalance as shown in the Result Summary.

Return values:

<GainImbalance> <numeric value>

Minimum, maximum or current Gain Imbalance, depending on the command syntax.

Default unit: dB

Example: `FETC:MACC:GIMB:MIN?`
would return, e.g.
`0.887`

Usage: Query only

Manual operation: See "Results to check modulation accuracy" on page 23

FETCh:MACCuracy:IQIMbalance:MAXimum[:RESult]?
FETCh:MACCuracy:IQIMbalance:MINimum[:RESult]?
FETCh:MACCuracy:IQIMbalance:CURRent[:RESult]?

This command queries the I/Q Imbalance as shown in the Result Summary.

Return values:

<IQImbalance> <numeric value>

Minimum, maximum or current I/Q Imbalance, depending on the command syntax.

Default unit: dB

Example: `FETC:MACC:IQIM:CURR?`
would return, e.g.
`0.02`

Usage: Query only

Manual operation: See "Results to check modulation accuracy" on page 23

FETCh:MACCuracy:IQOFfset:MAXimum[:RESult]?
FETCh:MACCuracy:IQOFfset:MINimum[:RESult]?
FETCh:MACCuracy:IQOFfset:CURRent[:RESult]?

This command queries the I/Q Offset as shown in the Result Summary.

Return values:

<IQOffset> <numeric value>

Minimum, maximum or current I/Q Offset, depending on the command syntax.

Default unit: dB

Example: `FETC:MACC:IQOF:MIN?`
would return, e.g.
`0.001`

Usage: Query only

Manual operation: See "Results to check modulation accuracy" on page 23

FETCh:MACCuracy:MERRor:MAXimum[:RESult]?
FETCh:MACCuracy:MERRor:MINimum[:RESult]?
FETCh:MACCuracy:MERRor:CURRent[:RESult]?

This command queries the Magnitude Error as shown in the Result Summary.

Return values:

<Magnitude> <numeric value>

Minimum, maximum or current Magnitude Error, depending on the command syntax.

Default unit: %

Example: `FETC:MACC:MERR:MAX?`
would return, e.g.
`1.12`

Usage: Query only

Manual operation: See "Results to check modulation accuracy" on page 23

FETCh:MACCuracy:PERRor:MAXimum[:RESult]?
FETCh:MACCuracy:PERRor:MINimum[:RESult]?
FETCh:MACCuracy:PERRor:CURRent[:RESult]?

This command queries the Phase Error as shown in the Result Summary.

Return values:

<PhaseError> <numeric value>

Minimum, maximum or current Phase Error, depending on the command syntax.

Default unit: degree

Example: `FETC:MACC:PERR:CURR?`
would return, e.g.
`1.84`

Usage: Query only

Manual operation: See "Results to check modulation accuracy" on page 23

FETCh:MACCuracy:QERRor:MAXimum[:RESult]?
FETCh:MACCuracy:QERRor:MINimum[:RESult]?
FETCh:MACCuracy:QERRor:CURRent[:RESult]?

This command queries the Quadrature Error as shown in the Result Summary.

Return values:

<QuadratureError> <numeric value>

Minimum, maximum or current Quadrature Error, depending on the command syntax.

Default unit: degree

Example: `FETC:MACC:QERR:MAX?`
would return, e.g.
`2.76`

Usage: Query only

Manual operation: See "Results to check modulation accuracy" on page 23

FETCh:MACCuracy:REVM:MAXimum[:RESult]?
FETCh:MACCuracy:REVM:MINimum[:RESult]?
FETCh:MACCuracy:REVM:CURRent[:RESult]?

This command queries the Raw EVM as shown in the Result Summary.

Return values:

<EVM> <numeric value>

Minimum, maximum or current Raw EVM, depending on the command syntax.

Default unit: %

Example: `FETC:MACC:REVM:MAX?`
would return, e.g.
`3.606`

Usage: Query only

Manual operation: See "Results to check modulation accuracy" on page 23

FETCh:MACCuracy:RMEV:MAXimum[:RESult]?
FETCh:MACCuracy:RMEV:MINimum[:RESult]?
FETCh:MACCuracy:RMEV:CURRent[:RESult]?

This command queries the Raw Model EVM as shown in the Result Summary.

Return values:

<EVM> <numeric value>

Minimum, maximum or current Raw Model EVM, depending on the command syntax.

Default unit: %

Example: `FETC:MACC:RMEV:CURR?`
would return, e.g.
`0.879`

Usage: Query only

Manual operation: See "Results to check modulation accuracy" on page 23

FETCh:MACCuracy:SRERror:MAXimum[:RESult]?
FETCh:MACCuracy:SRERror:MINimum[:RESult]?
FETCh:MACCuracy:SRERror:CURRent[:RESult]?

This command queries the Sample Rate Error as shown in the Result Summary.

Return values:

<SampleRateError> <numeric value>

Minimum, maximum or current SampleRateError, depending on the command syntax.

Default unit: Hz

Example: `FETC:MACC:SRER:CURR?`
would return, e.g.
`-0.023`

Usage: Query only

Manual operation: See "Results to check modulation accuracy" on page 23

Retrieving Power Results

FETCh:AMAM:CWIDth:CURRent[:RESult]?

This command queries the AM/AM Curve Width as shown in the Result Summary.

Return values:

<CurveWidth> <numeric value>

Current AM/AM Curve Width.

Default unit: dB

Example: `FETC:AMAM:CWID:CURR?`
would return, e.g.
`0.69`

Usage: Query only

Manual operation: See "Results to check power characteristics" on page 26

FETCh:AMPM:CWIDth:CURRent[:RESult]?

This command queries the AM/PM Curve Width as shown in the Result Summary.

Return values:

<CurveWidth> <numeric value>

Current AM/PM Curve Width.

Default unit: degree

Example: `FETC:AMPM:CWID:CURR?`
would return, e.g.
`1.441`

Usage: Query only

Manual operation: See "Results to check power characteristics" on page 26

FETCh:POWer:CFACtor:IN:CURRent[:RESult]?

This command queries the Crest Factor at the DUT input as shown in the Result Summary.

Return values:

<CrestFactor> <numeric value>

Current Crest Factor.

Default unit: dB

Example: `FETC:POW:CFAC:IN:CURR?`
would return, e.g.
`10.34`

Usage: Query only

Manual operation: See "Results to check power characteristics" on page 26

FETCh:POWer:CFACtor:OUT:CURRent[:RESult]?

This command queries the Crest Factor at the DUT output as shown in the Result Summary.

Return values:

<CrestFactor> <numeric value>

Current Crest Factor.

Default unit: dB

Example: `FETC:POW:CFAC:CURR?`
would return, e.g.
`8.72`

Usage: Query only

Manual operation: See "Results to check power characteristics" on page 26

FETCh:POWer:GAIN:MAXimum[:RESult]?
FETCh:POWer:GAIN:MINimum[:RESult]?
FETCh:POWer:GAIN:CURRent[:RESult]?

This command queries the signal gain as shown in the Result Summary.

Return values:

<Gain> — <numeric value>

Minimum, maximum or current gain, depending on the command syntax.

Default unit: dB

Example: `FETC:POW:GAIN:MAX?`
would return, e.g.
`21.37`

Usage: Query only

Manual operation: See "Results to check power characteristics" on page 26

FETCh:POWer:INPut:MAXimum[:RESult]?
FETCh:POWer:INPut:MINimum[:RESult]?
FETCh:POWer:INPut:CURRent[:RESult]?

This command queries the power at the DUT input as shown in the Result Summary.

Return values:

<Power> — <numeric value>

Minimum, maximum or current power, depending on the command syntax.

Default unit: dBm

Example: `FETC:POW:INP:MIN?`
would return, e.g.
`9.39`

Usage: Query only

Manual operation: See "Results to check power characteristics" on page 26

FETCh:POWer:OUTPut:MAXimum[:RESult]?
FETCh:POWer:OUTPut:MINimum[:RESult]?
FETCh:POWer:OUTPut:CURRent[:RESult]?

This command queries the signal power at the DUT output as shown in the Result Summary.

Return values:

<Power> <numeric value>

Minimum, maximum or current power, depending on the command syntax.

Default unit: dBm

Example: `FETC:POW:OUTP:MIN?`
would return, e.g.
`7.198`

Usage: Query only

Manual operation: See "Results to check power characteristics" on page 26

FETCh:POWer:P1DB:CURRent[:RESult]?

This command queries the 1 dB Compression Point as shown in the Result Summary.

Return values:

<Level> <numeric value>

Current 1 dB Compression Point.

Default unit: dBm

Example: `FETC:POW:P1DB:CURR?`
would return, e.g.
`-5.782`

Usage: Query only

Manual operation: See "Results to check power characteristics" on page 26

FETCh:POWer:P2DB:CURRent[:RESult]?

This command queries the 2 dB Compression Point as shown in the Result Summary.

Return values:

<Level> <numeric value>

Current 2 dB Compression Point.

Default unit: dBm

Example: `FETC:POW:P2DB:CURR?`
would return, e.g.
`-8.193`

Usage: Query only

Manual operation: See "Results to check power characteristics" on page 26

FETCh:POWer:P3DB:CURRent[:RESult]?

This command queries the 3 dB Compression Point as shown in the Result Summary.

Return values:

<Level>	<numeric value> Current 3 dB Compression Point. Default unit: dBm

Example:	`FETC:POW:P3DB:CURR?`
	would return, e.g.
	`2.551`

Usage:	Query only
Manual operation:	See "Results to check power characteristics" on page 26

FETCh:POWer:P1DB:MINimum[:RESult]

FETCh:POWer:P1DB:MAXimum[:RESult]

FETCh:POWer:P2DB:MINimum[:RESult]

FETCh:POWer:P2DB:MAXimum[:RESult]

FETCh:POWer:P3DB:MINimum[:RESult]

FETCh:POWer:P3DB:MAXimum[:RESult]

FETCh:POWer:OUTPut:P1DB:CURRent[:RESult]?

This command queries the output 1 dB Compression Point as shown in the Result Summary.

Return values:

<Level>	<numeric value> Current output 1 dB Compression Point. Default unit: dBm

Example:	`FETC:POW:OUTP:P1DB:CURR?`
	would return, e.g.
	`-5.782`

Usage:	Query only
Manual operation:	See "Results to check power characteristics" on page 26

FETCh:POWer:OUTPut:P2DB:CURRent[:RESult]?

This command queries the output 2 dB Compression Point as shown in the Result Summary.

Return values:

<Level> <numeric value>

Current output 2dB Compression Point.

Default unit: dBm

Example: `FETC:POW:OUTP:P2DB:CURR?`
would return, e.g.
`-5.782`

Usage: Query only

Manual operation: See "Results to check power characteristics" on page 26

FETCh:POWer:OUTPut:P3DB:CURRent[:RESult]?

This command queries the output 3 dB Compression Point as shown in the Result Summary.

Return values:

<Level> <numeric value>

Current output 3 dB Compression Point.

Default unit: dBm

Example: `FETC:POW:OUTP:P3DB:CURR?`
would return, e.g.
`-5.782`

Usage: Query only

Manual operation: See "Results to check power characteristics" on page 26

FETCh:POWer:OUTPut:P1DB:MINimum[:RESult]

FETCh:POWer:OUTPut:P1DB:MAXimum[:RESult]

FETCh:POWer:OUTPut:P2DB:MINimum[:RESult]

FETCh:POWer:OUTPut:P2DB:MAXimum[:RESult]

FETCh:POWer:OUTPut:P3DB:MINimum[:RESult]

FETCh:POWer:OUTPut:P3DB:MAXimum[:RESult]

6.5.3.3 Retrieving Results of the Parameter Sweep Table

Retrieving the results in the Parameter Sweep Table requires six commands for every result type.

Example command set to query the EVM results:

- `FETCh:PTABle:EVM:MAXimum[:RESult]` queries the highest EVM that has been measured.
- `FETCh:PTABle:EVM:MAXimum:X[:RESult]` queries the location on the x-axis where the highest EVM has been measured.
- `FETCh:PTABle:EVM:MAXimum:Y[:RESult]` queries the location on the y-axis where the highest EVM has been measured.
- `FETCh:PTABle:EVM:MINimum[:RESult]` queries the lowest EVM that has been measured.
- `FETCh:PTABle:EVM:MINimum:X[:RESult]` queries the location on the x-axis where the lowest EVM has been measured.
- `FETCh:PTABle:EVM:MINimum:Y[:RESult]` queries the location on the y-axis where the lowest EVM has been measured.

The type and unit of the value queried on the x- and y-axes depends on the parameter you have selected with `CONFigure:PSWeep:X:SETTing` and `CONFigure:PSWeep:Y:SETTing`.

FETCh:PTABle[:RESult]:ALL?

This command queries all numerical results shown in the Parameter Sweep Table.

Return values:

<Results> <numeric value>: Results as a comma separated list.
`<EVMMinValue>, <EVMMinX>, <EVMMinY>, <ACPMinCalue>, <ACPMinX>, <ACPMinY>,...`
The unit depends on the result and parameters assigned to the x- and y-axis.
If a result hasn't been calculated, the command returns `NAN`.

Example: `FETC:PTAB:ALL?`
would return, e.g.
`0.244445,1e+007,-30,0.246109,2e+007,-30,-21.9096,3e+007,-3`
`[etc.]`

Usage: Query only

FETCh:PTABle:ACP:ACHannel<ch>:BALanced:MAXimum:X[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:BALanced:MAXimum:Y[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:BALanced:MAXimum[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:BALanced:MINimum:X[:RESult]?

FETCh:PTABle:ACP:ACHannel<ch>:BALanced:MINimum:Y[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:BALanced:MINimum[:RESult]?

These commands query the result values for the ACP Balanced result as shown in the Parameter Sweep Table.

Suffix:

<ch>	1..n Window

Return values:

<Results>	<numeric value> • For `...[:RESult]`: Minimum or maximum result that has been measured. • For `...:X[:RESult]`: Location on the x-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the x-axis (`CONFigure:PSWeep:X:SETTing`). • For `...:Y[:RESult]`: Location on the y-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the y-axis (`CONFigure:PSWeep:Y:SETTing`).

Example:	`FETC:PTAB:ACP:ACH:BAL:MAX?`
	would return, e.g.
	`0.2[DB]`

Usage:	Query only

FETCh:PTABle:ACP:MAXimum:X[:RESult]?
FETCh:PTABle:ACP:MAXimum:Y[:RESult]?
FETCh:PTABle:ACP:MAXimum[:RESult]?
FETCh:PTABle:ACP:MINimum:X[:RESult]?
FETCh:PTABle:ACP:MINimum:Y[:RESult]?
FETCh:PTABle:ACP:MINimum[:RESult]?

These commands query the result values for the ACP result as shown in the Parameter Sweep Table.

Return values:

<Results>	<numeric value> • For `...[:RESult]`: Minimum or maximum result that has been measured. • For `...:X[:RESult]`: Location on the x-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the x-axis (`CONFigure:PSWeep:X:SETTing`). • For `...:Y[:RESult]`: Location on the y-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the y-axis (`CONFigure:PSWeep:Y:SETTing`).

Example: `FETC:PTAB:ACP:MAX?`
would return, e.g.
`-7.651[DBM]`

Usage: Query only

FETCh:PTABle:ACP:ACHannel<ch>:LOWer:MAXimum:X[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:LOWer:MAXimum:Y[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:LOWer:MAXimum[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:LOWer:MINimum:X[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:LOWer:MINimum:Y[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:LOWer:MINimum[:RESult]?

These commands query the result values for the lower adjacent channel power as shown in the Parameter Sweep Table.

Suffix:
<ch> 1..n
irrelevant

Return values:
<Results> <numeric value>

• For `...[:RESult]`: Minimum or maximum result that has been measured.
• For `...:X[:RESult]`: Location on the x-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the x-axis (`CONFigure:PSWeep:X:SETTing`).
• For `...:Y[:RESult]`: Location on the y-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the y-axis (`CONFigure:PSWeep:Y:SETTing`).

Example: `FETC:PTAB:ACP:ACH:LOW:MIN?`
would return, e.g.
`-10.945[DBM]`

Usage: Query only

FETCh:PTABle:ACP:ACHannel<ch>:UPPer:MAXimum:X[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:UPPer:MAXimum:Y[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:UPPer:MAXimum[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:UPPer:MINimum:X[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:UPPer:MINimum:Y[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:UPPer:MINimum[:RESult]?

These commands query the result values for the upper adjacent channel power as shown in the Parameter Sweep Table.

Suffix:
<ch> 1..n
irrelevant

Return values:

<Results> — <numeric value>

• For `...[:RESult]`: Minimum or maximum result that has been measured.
• For `...:X[:RESult]`: Location on the x-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the x-axis (`CONFigure:PSWeep:X:SETTing`).
• For `...:Y[:RESult]`: Location on the y-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the y-axis (`CONFigure:PSWeep:Y:SETTing`).

Example: `FETC:PTAB:ACP:ACH:UPP:MIN:Y?`
would return, e.g. (if the y-axis represents the frequency)
`10000000[HZ]`

Usage: Query only

FETCh:PTABle:AMAM:CWIDth:MAXimum:X[:RESult]?
FETCh:PTABle:AMAM:CWIDth:MAXimum:Y[:RESult]?
FETCh:PTABle:AMAM:CWIDth:MAXimum[:RESult]?
FETCh:PTABle:AMAM:CWIDth:MINimum:X[:RESult]?
FETCh:PTABle:AMAM:CWIDth:MINimum:Y[:RESult]?
FETCh:PTABle:AMAM:CWIDth:MINimum[:RESult]?

These commands query the result values for the AM/AM Curve Width result as shown in the Parameter Sweep Table.

Return values:

<Results> — <numeric value>

• For `...[:RESult]`: Minimum or maximum result that has been measured.
• For `...:X[:RESult]`: Location on the x-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the x-axis (`CONFigure:PSWeep:X:SETTing`).
• For `...:Y[:RESult]`: Location on the y-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the y-axis (`CONFigure:PSWeep:Y:SETTing`).

Example: `FETC:PTAB:AMAM:CWID:MAX:X?`
would return, e.g. (if the x-axis represents the RF to envelope delay)
`-0.000001[s]`

Usage: Query only

FETCh:PTABle:AMPM:CWIDth:MAXimum:X[:RESult]?
FETCh:PTABle:AMPM:CWIDth:MAXimum:Y[:RESult]?
FETCh:PTABle:AMPM:CWIDth:MAXimum[:RESult]?
FETCh:PTABle:AMPM:CWIDth:MINimum:X[:RESult]?
FETCh:PTABle:AMPM:CWIDth:MINimum:Y[:RESult]?
FETCh:PTABle:AMPM:CWIDth:MINimum[:RESult]?

These commands query the result values for the AM/PM Curve Width result as shown in the Parameter Sweep Table.

Return values:

<Results> <numeric value>

• For `...[:RESult]`: Minimum or maximum result that has been measured.
• For `...:X[:RESult]`: Location on the x-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the x-axis (`CONFigure:PSWeep:X:SETTing`).
• For `...:Y[:RESult]`: Location on the y-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the y-axis (`CONFigure:PSWeep:Y:SETTing`).

Example: `FETC:PTAB:AMPM:CWID:MAX:X?`
would return, e.g. (if the x-axis represents the frequency)
`150000000[HZ]`

Usage: Query only

FETCh:PTABle:CFACtor:MAXimum:X[:RESult]?
FETCh:PTABle:CFACtor:MAXimum:Y[:RESult]?
FETCh:PTABle:CFACtor:MAXimum[:RESult]?
FETCh:PTABle:CFACtor:MINimum:X[:RESult]?
FETCh:PTABle:CFACtor:MINimum:Y[:RESult]?
FETCh:PTABle:CFACtor:MINimum[:RESult]?

These commands query the result values for the Crest Factor result as shown in the Parameter Sweep Table.

Return values:

<Results> <numeric value>

• For `...[:RESult]`: Minimum or maximum result that has been measured.
• For `...:X[:RESult]`: Location on the x-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the x-axis (`CONFigure:PSWeep:X:SETTing`).
• For `...:Y[:RESult]`: Location on the y-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the y-axis (`CONFigure:PSWeep:Y:SETTing`).

Example: `FETC:PTAB:CFAC:MIN?`
would return, e.g.
`0.053[DB]`

Usage: Query only

FETCh:PTABle:EVM:MAXimum:X[:RESult]?
FETCh:PTABle:EVM:MAXimum:Y[:RESult]?
FETCh:PTABle:EVM:MAXimum[:RESult]?
FETCh:PTABle:EVM:MINimum:X[:RESult]?
FETCh:PTABle:EVM:MINimum:Y[:RESult]?
FETCh:PTABle:EVM:MINimum[:RESult]?

These commands query the result values for the EVM result as shown in the Parameter Sweep Table.

Return values:
<Results> <numeric value>

• For `...[:RESult]`: Minimum or maximum result that has been measured.
• For `...:X[:RESult]`: Location on the x-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the x-axis (`CONFigure:PSWeep:X:SETTing`).
• For `...:Y[:RESult]`: Location on the y-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the y-axis (`CONFigure:PSWeep:Y:SETTing`).

Example: `FETC:PTAB:EVM:MAX:Y?`
would return, e.g. (if the y-axis represents the output power)
`0[DBM]`

Usage: Query only

FETCh:PTABle:GAIN:MAXimum:X[:RESult]?
FETCh:PTABle:GAIN:MAXimum:Y[:RESult]?
FETCh:PTABle:GAIN:MAXimum[:RESult]?
FETCh:PTABle:GAIN:MINimum:X[:RESult]?
FETCh:PTABle:GAIN:MINimum:Y[:RESult]?
FETCh:PTABle:GAIN:MINimum[:RESult]?

These commands query the result values for the Gain result as shown in the Parameter Sweep Table.

Return values:

<Results> <numeric value>

• For `...[:RESult]`: Minimum or maximum result that has been measured.
• For `...:X[:RESult]`: Location on the x-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the x-axis (`CONFigure:PSWeep:X:SETTing`).
• For `...:Y[:RESult]`: Location on the y-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the y-axis (`CONFigure:PSWeep:Y:SETTing`).

Example: `FETC:PTAB:GAIN:MAX?`
would return, e.g.
`-5.392[DBM]`

Usage: Query only

FETCh:PTABle:P1DB:MAXimum:X[:RESult]?
FETCh:PTABle:P1DB:MAXimum:Y[:RESult]?
FETCh:PTABle:P1DB:MAXimum[:RESult]?
FETCh:PTABle:P1DB:MINimum:X[:RESult]?
FETCh:PTABle:P1DB:MINimum:Y[:RESult]?
FETCh:PTABle:P1DB:MINimum[:RESult]?

These commands query the result values for the 1 dB Compression Point result as shown in the Parameter Sweep Table.

Return values:

<Results> <numeric value>

• For `...[:RESult]`: Minimum or maximum result that has been measured.
• For `...:X[:RESult]`: Location on the x-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the x-axis (`CONFigure:PSWeep:X:SETTing`).
• For `...:Y[:RESult]`: Location on the y-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the y-axis (`CONFigure:PSWeep:Y:SETTing`).

Example: `FETC:PTAB:P1DB:MIN?`
would return, e.g.
`-9.382[DBM]`

Usage: Query only

FETCh:PTABle:P2DB:MAXimum:X[:RESult]?
FETCh:PTABle:P2DB:MAXimum:Y[:RESult]?

FETCh:PTABle:P2DB:MAXimum[:RESult]?
FETCh:PTABle:P2DB:MINimum:X[:RESult]?
FETCh:PTABle:P2DB:MINimum:Y[:RESult]?
FETCh:PTABle:P2DB:MINimum[:RESult]?

These commands query the result values for the 2 dB Compression Point result as shown in the Parameter Sweep Table.

Return values:

<Results> <numeric value>

• For `...[:RESult]`: Minimum or maximum result that has been measured.
• For `...:X[:RESult]`: Location on the x-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the x-axis (`CONFigure:PSWeep:X:SETTing`).
• For `...:Y[:RESult]`: Location on the y-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the y-axis (`CONFigure:PSWeep:Y:SETTing`).

Example: `FETC:PTAB:P2DB:MIN?`
would return, e.g.
`-4.229[DBM]`

Usage: Query only

FETCh:PTABle:P3DB:MAXimum:X[:RESult]?
FETCh:PTABle:P3DB:MAXimum:Y[:RESult]?
FETCh:PTABle:P3DB:MAXimum[:RESult]?
FETCh:PTABle:P3DB:MINimum:X[:RESult]?
FETCh:PTABle:P3DB:MINimum:Y[:RESult]?
FETCh:PTABle:P3DB:MINimum[:RESult]?

These commands query the result values for the 3 dB Compression Point result as shown in the Parameter Sweep Table.

Return values:

<Results> <numeric value>

• For `...[:RESult]`: Minimum or maximum result that has been measured.
• For `...:X[:RESult]`: Location on the x-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the x-axis (`CONFigure:PSWeep:X:SETTing`).
• For `...:Y[:RESult]`: Location on the y-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the y-axis (`CONFigure:PSWeep:Y:SETTing`).

Example: `FETC:PTAB:P3DB:MIN?`
would return, e.g.
`-12.038[DBM]`

Usage: Query only

FETCh:PTABle:POUT:MAXimum:X[:RESult]?
FETCh:PTABle:POUT:MAXimum:Y[:RESult]?
FETCh:PTABle:POUT:MAXimum[:RESult]?
FETCh:PTABle:POUT:MINimum:X[:RESult]?
FETCh:PTABle:POUT:MINimum:Y[:RESult]?
FETCh:PTABle:POUT:MINimum[:RESult]?

These commands query the result values for the Power Out result as shown in the Parameter Sweep Table.

Return values:

<Results> <numeric value>

• For `...[:RESult]`: Minimum or maximum result that has been measured.
• For `...:X[:RESult]`: Location on the x-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the x-axis (`CONFigure:PSWeep:X:SETTing`).
• For `...:Y[:RESult]`: Location on the y-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the y-axis (`CONFigure:PSWeep:Y:SETTing`).

Example: `FETC:PTAB:POUT:MIN?`
would return, e.g.
`-12.032[DBM]`

Usage: Query only

FETCh:PTABle:RMS:MAXimum:X[:RESult]?
FETCh:PTABle:RMS:MAXimum:Y[:RESult]?
FETCh:PTABle:RMS:MAXimum[:RESult]?
FETCh:PTABle:RMS:MINimum:X[:RESult]?
FETCh:PTABle:RMS:MINimum:Y[:RESult]?
FETCh:PTABle:RMS:MINimum[:RESult]?

These commands query the result values for the RMS Power result as shown in the Parameter Sweep Table.

Return values:

<Results> <numeric value>

• For `...[:RESult]`: Minimum or maximum result that has been measured.
• For `...:X[:RESult]`: Location on the x-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the x-axis (`CONFigure:PSWeep:X:SETTing`).
• For `...:Y[:RESult]`: Location on the y-axis where the minimum or maximum result has been measured. The type of value depends on the parameter you have selected for the y-axis (`CONFigure:PSWeep:Y:SETTing`).

Example: `FETC:PTAB:RMS:MIN?`
would return, e.g.
`-12.032[DBM]`

Usage: Query only

6.5.3.4 Retrieving Results of the Statistics Table

Retrieving the results in the statistics table requires different commands for every result type.

In the following example, the SCPI commands querying the statistical results for amplitude droop are described. The argument "ALL" is used to retrieve the results over the entire measurement and "CURRent" is used to retrieve results over the current capture.

- `FETCh:STABle:ADRoop:AVERage? ALL` queries the statistical value of all result ranges found in the entire measurement (corresponding to black area of statistics table).
- `FETCh:STABle:ADRoop:AVERage? CURRent` queries the statistical value of all result ranges found in the current capture (corresponding to green area of statistics table).
- `FETCh:STABle:ADRoop:SELected[:RESult?]` queries the result of the currently selected result range (corresponds to blue area of statistics table).

Amplitude Droop

FETCh:STABle:ADRoop:AVERage? ALL

FETCh:STABle:ADRoop:AVERage? CURRent

FETCh:STABle:ADRoop:MAXimum? ALL

FETCh:STABle:ADRoop:MAXimum? CURRent

FETCh:STABle:ADRoop:MINimum? ALL

FETCh:STABle:ADRoop:MINimum? CURRent

FETCh:STABle:ADRoop:SELected[:RESult?]

FETCh:STABle:ADRoop:STDeviation? ALL

FETCh:STABle:ADRoop:STDeviation? CURRent

AM/AM Curve Width

FETCh:STABle:AMAM:CWIDth:AVERage? ALL

FETCh:STABle:AMAM:CWIDth:AVERage? CURRent

FETCh:STABle:AMAM:CWIDth:MAXimum? ALL

FETCh:STABle:AMAM:CWIDth:MAXimum? CURRent

FETCh:STABle:AMAM:CWIDth:MINimum? ALL

FETCh:STABle:AMAM:CWIDth:MINimum? CURRent

FETCh:STABle:AMAM:CWIDth:SELected[:RESult?]

FETCh:STABle:AMAM:CWIDth:STDeviation? ALL

FETCh:STABle:AMAM:CWIDth:STDeviation? CURRent

AM/PM Curve Width

FETCh:STABle:AMPM:CWIDth:AVERage? ALL

FETCh:STABle:AMPM:CWIDth:AVERage? CURRent

FETCh:STABle:AMPM:CWIDth:MAXimum? ALL

FETCh:STABle:AMPM:CWIDth:MAXimum? CURRent

FETCh:STABle:AMPM:CWIDth:MINimum? ALL

FETCh:STABle:AMPM:CWIDth:MINimum? CURRent

FETCh:STABle:AMPM:CWIDth:SELected[:RESult?]

FETCh:STABle:AMPM:CWIDth:STDeviation? ALL

FETCh:STABle:AMPM:CWIDth:STDeviation? CURRent

Average PAE

FETCh:STABle:APAE:AVG:AVERage? ALL

FETCh:STABle:APAE:AVG:AVERage? CURRent

FETCh:STABle:APAE:AVG:MAXimum? ALL

FETCh:STABle:APAE:AVG:MAXimum? CURRent

FETCh:STABle:APAE:AVG:MINimum? ALL

FETCh:STABle:APAE:AVG:MINimum? CURRent

FETCh:STABle:APAE:AVG:SELected[:RESult?]

FETCh:STABle:APAE:AVG:STDeviation? ALL

FETCh:STABle:APAE:AVG:STDeviation? CURRent

FETCh:STABle:APAE:MAX:AVERage? ALL

FETCh:STABle:APAE:MAX:AVERage? CURRent

FETCh:STABle:APAE:MAX:MAXimum? ALL

FETCh:STABle:APAE:MAX:MAXimum? CURRent

FETCh:STABle:APAE:MAX:MINimum? ALL

FETCh:STABle:APAE:MAX:MINimum? CURRent

FETCh:STABle:APAE:MAX:SELected[:RESult?]

FETCh:STABle:APAE:MAX:STDeviation? ALL

FETCh:STABle:APAE:MAX:STDeviation? CURRent

FETCh:STABle:APAE:MIN:AVERage? ALL

FETCh:STABle:APAE:MIN:AVERage? CURRent

FETCh:STABle:APAE:MIN:MAXimum? ALL

FETCh:STABle:APAE:MIN:MAXimum? CURRent

FETCh:STABle:APAE:MIN:MINimum? ALL

FETCh:STABle:APAE:MIN:MINimum? CURRent

FETCh:STABle:APAE:MIN:SELected[:RESult?]

FETCh:STABle:APAE:MIN:STDeviation? ALL

FETCh:STABle:APAE:MIN:STDeviation? CURRent

Power (Vcc*Icc)

FETCh:STABle:BBPower:AVG:AVERage? ALL

FETCh:STABle:BBPower:AVG:AVERage? CURRent

FETCh:STABle:BBPower:AVG:MAXimum? ALL

FETCh:STABle:BBPower:AVG:MAXimum? CURRent

FETCh:STABle:BBPower:AVG:MINimum? ALL

FETCh:STABle:BBPower:AVG:MINimum? CURRent

FETCh:STABle:BBPower:AVG:SELected[:RESult?]

FETCh:STABle:BBPower:AVG:STDeviation? ALL

FETCh:STABle:BBPower:AVG:STDeviation? CURRent

FETCh:STABle:BBPower:MAX:AVERage? ALL

FETCh:STABle:BBPower:MAX:AVERage? CURRent

FETCh:STABle:BBPower:MAX:MAXimum? ALL

FETCh:STABle:BBPower:MAX:MAXimum? CURRent

FETCh:STABle:BBPower:MAX:MINimum? ALL

FETCh:STABle:BBPower:MAX:MINimum? CURRent

FETCh:STABle:BBPower:MAX:SELected[:RESult?]

FETCh:STABle:BBPower:MAX:STDeviation? ALL

FETCh:STABle:BBPower:MAX:STDeviation? CURRent

FETCh:STABle:BBPower:MIN:AVERage? ALL

FETCh:STABle:BBPower:MIN:AVERage? CURRent

FETCh:STABle:BBPower:MIN:MAXimum? ALL

FETCh:STABle:BBPower:MIN:MAXimum? CURRent

FETCh:STABle:BBPower:MIN:MINimum? ALL

FETCh:STABle:BBPower:MIN:MINimum? CURRent

FETCh:STABle:BBPower:MIN:SELected[:RESult?]

FETCh:STABle:BBPower:MIN:STDeviation? ALL

FETCh:STABle:BBPower:MIN:STDeviation? CURRent

Crest Factor Input

FETCh:STABle:CFACtor:IN:AVERage? ALL

FETCh:STABle:CFACtor:IN:AVERage? CURRent

FETCh:STABle:CFACtor:IN:MAXimum? ALL

FETCh:STABle:CFACtor:IN:MAXimum? CURRent

FETCh:STABle:CFACtor:IN:MINimum? ALL

FETCh:STABle:CFACtor:IN:MINimum? CURRent

FETCh:STABle:CFACtor:IN:SELected[:RESult?]

FETCh:STABle:CFACtor:IN:STDeviation? ALL

FETCh:STABle:CFACtor:IN:STDeviation? CURRent

Crest Factor Out

FETCh:STABle:CFACtor:OUT:AVERage? ALL

FETCh:STABle:CFACtor:OUT:AVERage? CURRent

FETCh:STABle:CFACtor:OUT:MAXimum? ALL

FETCh:STABle:CFACtor:OUT:MAXimum? CURRent

FETCh:STABle:CFACtor:OUT:MINimum? ALL

FETCh:STABle:CFACtor:OUT:MINimum? CURRent

FETCh:STABle:CFACtor:OUT:SELected[:RESult?]

FETCh:STABle:CFACtor:OUT:STDeviation? ALL

FETCh:STABle:CFACtor:OUT:STDeviation? CURRent

Frequency Error

FETCh:STABle:FERRor:AVERage? ALL

FETCh:STABle:FERRor:AVERage? CURRent

FETCh:STABle:FERRor:MAXimum? ALL

FETCh:STABle:FERRor:MAXimum? CURRent

FETCh:STABle:FERRor:MINimum? ALL

FETCh:STABle:FERRor:MINimum? CURRent

FETCh:STABle:FERRor:SELected[:RESult?]

FETCh:STABle:FERRor:STDeviation? ALL

FETCh:STABle:FERRor:STDeviation? CURRent

Gain

FETCh:STABle:GAIN:AVERage? ALL

FETCh:STABle:GAIN:AVERage? CURRent

FETCh:STABle:GAIN:MAXimum? ALL

FETCh:STABle:GAIN:MAXimum? CURRent

FETCh:STABle:GAIN:MINimum? ALL

FETCh:STABle:GAIN:MINimum? CURRent

FETCh:STABle:GAIN:SELected[:RESult?]

FETCh:STABle:GAIN:STDeviation? ALL

FETCh:STABle:GAIN:STDeviation? CURRent

Gain Imbalance

FETCh:STABle:GIMBalance:AVERage? ALL

FETCh:STABle:GIMBalance:AVERage? CURRent

FETCh:STABle:GIMBalance:MAXimum? ALL

FETCh:STABle:GIMBalance:MAXimum? CURRent

FETCh:STABle:GIMBalance:MINimum? ALL

FETCh:STABle:GIMBalance:MINimum? CURRent

FETCh:STABle:GIMBalance:SELected[:RESult?]

FETCh:STABle:GIMBalance:STDeviation? ALL

FETCh:STABle:GIMBalance:STDeviation? CURRent

Current (Icc)

FETCh:STABle:ICC:AVG:AVERage? ALL

FETCh:STABle:ICC:AVG:AVERage? CURRent

FETCh:STABle:ICC:AVG:MAXimum? ALL

FETCh:STABle:ICC:AVG:MAXimum? CURRent

FETCh:STABle:ICC:AVG:MINimum? ALL

FETCh:STABle:ICC:AVG:MINimum? CURRent

FETCh:STABle:ICC:AVG:SELected[:RESult?]

FETCh:STABle:ICC:AVG:STDeviation? ALL

FETCh:STABle:ICC:AVG:STDeviation? CURRent

FETCh:STABle:ICC:MAX:AVERage? ALL

FETCh:STABle:ICC:MAX:AVERage? CURRent

FETCh:STABle:ICC:MAX:MAXimum? ALL

FETCh:STABle:ICC:MAX:MAXimum? CURRent

FETCh:STABle:ICC:MAX:MINimum? ALL

FETCh:STABle:ICC:MAX:MINimum? CURRent

FETCh:STABle:ICC:MAX:SELected[:RESult?]

FETCh:STABle:ICC:MAX:STDeviation? ALL

FETCh:STABle:ICC:MAX:STDeviation? CURRent

FETCh:STABle:ICC:MIN:AVERage? ALL

FETCh:STABle:ICC:MIN:AVERage? CURRent

FETCh:STABle:ICC:MIN:MAXimum? ALL

FETCh:STABle:ICC:MIN:MAXimum? CURRent

FETCh:STABle:ICC:MIN:MINimum? ALL

FETCh:STABle:ICC:MIN:MINimum? CURRent

FETCh:STABle:ICC:MIN:SELected[:RESult?]

FETCh:STABle:ICC:MIN:STDeviation? ALL

FETCh:STABle:ICC:MIN:STDeviation? CURRent

I/Q Imbalance

FETCh:STABle:IQIMbalance:AVERage? ALL

FETCh:STABle:IQIMbalance:AVERage? CURRent

FETCh:STABle:IQIMbalance:MAXimum? ALL

FETCh:STABle:IQIMbalance:MAXimum? CURRent

FETCh:STABle:IQIMbalance:MINimum? ALL

FETCh:STABle:IQIMbalance:MINimum? CURRent

FETCh:STABle:IQIMbalance:SELected[:RESult?]

FETCh:STABle:IQIMbalance:STDeviation? ALL

FETCh:STABle:IQIMbalance:STDeviation? CURRent

I/Q Offset

FETCh:STABle:IQOFfset:AVERage? ALL

FETCh:STABle:IQOFfset:AVERage? CURRent

FETCh:STABle:IQOFfset:MAXimum? ALL

FETCh:STABle:IQOFfset:MAXimum? CURRent

FETCh:STABle:IQOFfset:MINimum? ALL

FETCh:STABle:IQOFfset:MINimum? CURRent

FETCh:STABle:IQOFfset:SELected[:RESult?]

FETCh:STABle:IQOFfset:STDeviation? ALL

FETCh:STABle:IQOFfset:STDeviation? CURRent

Baseband I Input Voltage

FETCh:STABle:IVOLtage:AVG:AVERage? ALL

FETCh:STABle:IVOLtage:AVG:AVERage? CURRent

FETCh:STABle:IVOLtage:AVG:MAXimum? ALL

FETCh:STABle:IVOLtage:AVG:MAXimum? CURRent

FETCh:STABle:IVOLtage:AVG:MINimum? ALL

FETCh:STABle:IVOLtage:AVG:MINimum? CURRent

FETCh:STABle:IVOLtage:AVG:SELected[:RESult?]

FETCh:STABle:IVOLtage:AVG:STDeviation? ALL

FETCh:STABle:IVOLtage:AVG:STDeviation? CURRent

FETCh:STABle:IVOLtage:MAX:AVERage? ALL

FETCh:STABle:IVOLtage:MAX:AVERage? CURRent

FETCh:STABle:IVOLtage:MAX:MAXimum? ALL

FETCh:STABle:IVOLtage:MAX:MAXimum? CURRent

FETCh:STABle:IVOLtage:MAX:MINimum? ALL

FETCh:STABle:IVOLtage:MAX:MINimum? CURRent

FETCh:STABle:IVOLtage:MAX:SELected[:RESult?]

FETCh:STABle:IVOLtage:MAX:STDeviation? ALL

FETCh:STABle:IVOLtage:MAX:STDeviation? CURRent

FETCh:STABle:IVOLtage:MIN:AVERage? ALL

FETCh:STABle:IVOLtage:MIN:AVERage? CURRent

FETCh:STABle:IVOLtage:MIN:MAXimum? ALL

FETCh:STABle:IVOLtage:MIN:MAXimum? CURRent

FETCh:STABle:IVOLtage:MIN:MINimum? ALL

FETCh:STABle:IVOLtage:MIN:MINimum? CURRent

FETCh:STABle:IVOLtage:MIN:SELected[:RESult?]

FETCh:STABle:IVOLtage:MIN:STDeviation? ALL

FETCh:STABle:IVOLtage:MIN:STDeviation? CURRent

Magnitude Error

FETCh:STABle:MERRor:AVERage? ALL

FETCh:STABle:MERRor:AVERage? CURRent

FETCh:STABle:MERRor:MAXimum? ALL

FETCh:STABle:MERRor:MAXimum? CURRent

FETCh:STABle:MERRor:MINimum? ALL

FETCh:STABle:MERRor:MINimum? CURRent

FETCh:STABle:MERRor:SELected[:RESult?]

FETCh:STABle:MERRor:STDeviation? ALL

FETCh:STABle:MERRor:STDeviation? CURRent

P1DB Input

FETCh:STABle:P1DB:IN:AVERage? ALL

FETCh:STABle:P1DB:IN:AVERage? CURRent

FETCh:STABle:P1DB:IN:MAXimum? ALL

FETCh:STABle:P1DB:IN:MAXimum? CURRent

FETCh:STABle:P1DB:IN:MINimum? ALL

FETCh:STABle:P1DB:IN:MINimum? CURRent

FETCh:STABle:P1DB:IN:SELected[:RESult?]

FETCh:STABle:P1DB:IN:STDeviation? ALL

FETCh:STABle:P1DB:IN:STDeviation? CURRent

P1DB Output

FETCh:STABle:P1DB:OUT:AVERage? ALL

FETCh:STABle:P1DB:OUT:AVERage? CURRent

FETCh:STABle:P1DB:OUT:MAXimum? ALL

FETCh:STABle:P1DB:OUT:MAXimum? CURRent

FETCh:STABle:P1DB:OUT:MINimum? ALL

FETCh:STABle:P1DB:OUT:MINimum? CURRent

FETCh:STABle:P1DB:OUT:SELected[:RESult?]

FETCh:STABle:P1DB:OUT:STDeviation? ALL

FETCh:STABle:P1DB:OUT:STDeviation? CURRent

P2DB Input

FETCh:STABle:P2DB:IN:AVERage? ALL

FETCh:STABle:P2DB:IN:AVERage? CURRent

FETCh:STABle:P2DB:IN:MAXimum? ALL

FETCh:STABle:P2DB:IN:MAXimum? CURRent

FETCh:STABle:P2DB:IN:MINimum? ALL

FETCh:STABle:P2DB:IN:MINimum? CURRent

FETCh:STABle:P2DB:IN:SELected[:RESult?]

FETCh:STABle:P2DB:IN:STDeviation? ALL

FETCh:STABle:P2DB:IN:STDeviation? CURRent

P2DB Output

FETCh:STABle:P2DB:OUT:AVERage? ALL

FETCh:STABle:P2DB:OUT:AVERage? CURRent

FETCh:STABle:P2DB:OUT:MAXimum? ALL

FETCh:STABle:P2DB:OUT:MAXimum? CURRent

FETCh:STABle:P2DB:OUT:MINimum? ALL

FETCh:STABle:P2DB:OUT:MINimum? CURRent

FETCh:STABle:P2DB:OUT:SELected[:RESult?]

FETCh:STABle:P2DB:OUT:STDeviation? ALL

FETCh:STABle:P2DB:OUT:STDeviation? CURRent

P3DB Input

FETCh:STABle:P3DB:IN:AVERage? ALL

FETCh:STABle:P3DB:IN:AVERage? CURRent

FETCh:STABle:P3DB:IN:MAXimum? ALL

FETCh:STABle:P3DB:IN:MAXimum? CURRent

FETCh:STABle:P3DB:IN:MINimum? ALL

FETCh:STABle:P3DB:IN:MINimum? CURRent

FETCh:STABle:P3DB:IN:SELected[:RESult?]

FETCh:STABle:P3DB:IN:STDeviation? ALL

FETCh:STABle:P3DB:IN:STDeviation? CURRent

P3DB Output

FETCh:STABle:P3DB:OUT:AVERage? ALL

FETCh:STABle:P3DB:OUT:AVERage? CURRent

FETCh:STABle:P3DB:OUT:MAXimum? ALL

FETCh:STABle:P3DB:OUT:MAXimum? CURRent

FETCh:STABle:P3DB:OUT:MINimum? ALL

FETCh:STABle:P3DB:OUT:MINimum? CURRent

FETCh:STABle:P3DB:OUT:SELected[:RESult?]

FETCh:STABle:P3DB:OUT:STDeviation? ALL

FETCh:STABle:P3DB:OUT:STDeviation? CURRent

Average Power Consumption

FETCh:STABle:PC:AVERage? ALL

FETCh:STABle:PC:AVERage? CURRent

FETCh:STABle:PC:MAXimum? ALL

FETCh:STABle:PC:MAXimum? CURRent

FETCh:STABle:PC:MINimum? ALL

FETCh:STABle:PC:MINimum? CURRent

FETCh:STABle:PC:SELected[:RESult?]

FETCh:STABle:PC:STDeviation? ALL

FETCh:STABle:PC:STDeviation? CURRent

PC based Average PAE

FETCh:STABle:PCPA:AVERage? ALL

FETCh:STABle:PCPA:AVERage? CURRent

FETCh:STABle:PCPA:MAXimum? ALL

FETCh:STABle:PCPA:MAXimum? CURRent

FETCh:STABle:PCPA:MINimum? ALL

FETCh:STABle:PCPA:MINimum? CURRent

FETCh:STABle:PCPA:SELected[:RESult?]

FETCh:STABle:PCPA:STDeviation? ALL

FETCh:STABle:PCPA:STDeviation? CURRent

Phase Error

FETCh:STABle:PERRor:AVERage? ALL

FETCh:STABle:PERRor:AVERage? CURRent

FETCh:STABle:PERRor:MAXimum? ALL

FETCh:STABle:PERRor:MAXimum? CURRent

FETCh:STABle:PERRor:MINimum? ALL

FETCh:STABle:PERRor:MINimum? CURRent

FETCh:STABle:PERRor:SELected[:RESult?]

FETCh:STABle:PERRor:STDeviation? ALL

FETCh:STABle:PERRor:STDeviation? CURRent

Power Input

FETCh:STABle:POWer:INPut:AVG:AVERage? ALL

FETCh:STABle:POWer:INPut:AVG:AVERage? CURRent

FETCh:STABle:POWer:INPut:AVG:MAXimum? ALL

FETCh:STABle:POWer:INPut:AVG:MAXimum? CURRent

FETCh:STABle:POWer:INPut:AVG:MINimum? ALL

FETCh:STABle:POWer:INPut:AVG:MINimum? CURRent

FETCh:STABle:POWer:INPut:AVG:SELected[:RESult?]

FETCh:STABle:POWer:INPut:AVG:STDeviation? ALL

FETCh:STABle:POWer:INPut:AVG:STDeviation? CURRent

FETCh:STABle:POWer:INPut:MAX:AVERage? ALL

FETCh:STABle:POWer:INPut:MAX:AVERage? CURRent

FETCh:STABle:POWer:INPut:MAX:MAXimum? ALL

FETCh:STABle:POWer:INPut:MAX:MAXimum? CURRent

FETCh:STABle:POWer:INPut:MAX:MINimum? ALL

FETCh:STABle:POWer:INPut:MAX:MINimum? CURRent

FETCh:STABle:POWer:INPut:MAX:SELected[:RESult?]

FETCh:STABle:POWer:INPut:MAX:STDeviation? ALL

FETCh:STABle:POWer:INPut:MAX:STDeviation? CURRent

FETCh:STABle:POWer:INPut:MIN:AVERage? ALL

FETCh:STABle:POWer:INPut:MIN:AVERage? CURRent

FETCh:STABle:POWer:INPut:MIN:MAXimum? ALL

FETCh:STABle:POWer:INPut:MIN:MAXimum? CURRent

FETCh:STABle:POWer:INPut:MIN:MINimum? ALL

FETCh:STABle:POWer:INPut:MIN:MINimum? CURRent

FETCh:STABle:POWer:INPut:MIN:SELected[:RESult?]

FETCh:STABle:POWer:INPut:MIN:STDeviation? ALL

FETCh:STABle:POWer:INPut:MIN:STDeviation? CURRent

Power Output

FETCh:STABle:POWer:OUTPut:AVG:AVERage? ALL

FETCh:STABle:POWer:OUTPut:AVG:AVERage? CURRent

FETCh:STABle:POWer:OUTPut:AVG:MAXimum? ALL

FETCh:STABle:POWer:OUTPut:AVG:MAXimum? CURRent

FETCh:STABle:POWer:OUTPut:AVG:MINimum? ALL

FETCh:STABle:POWer:OUTPut:AVG:MINimum? CURRent

FETCh:STABle:POWer:OUTPut:AVG:SELected[:RESult?]

FETCh:STABle:POWer:OUTPut:AVG:STDeviation? ALL

FETCh:STABle:POWer:OUTPut:AVG:STDeviation? CURRent

FETCh:STABle:POWer:OUTPut:MAX:AVERage? ALL

FETCh:STABle:POWer:OUTPut:MAX:AVERage? CURRent

FETCh:STABle:POWer:OUTPut:MAX:MAXimum? ALL

FETCh:STABle:POWer:OUTPut:MAX:MAXimum? CURRent

FETCh:STABle:POWer:OUTPut:MAX:MINimum? ALL

FETCh:STABle:POWer:OUTPut:MAX:MINimum? CURRent

FETCh:STABle:POWer:OUTPut:MAX:SELected[:RESult?]

FETCh:STABle:POWer:OUTPut:MAX:STDeviation? ALL

FETCh:STABle:POWer:OUTPut:MAX:STDeviation? CURRent

FETCh:STABle:POWer:OUTPut:MIN:AVERage? ALL

FETCh:STABle:POWer:OUTPut:MIN:AVERage? CURRent

FETCh:STABle:POWer:OUTPut:MIN:MAXimum? ALL

FETCh:STABle:POWer:OUTPut:MIN:MAXimum? CURRent

FETCh:STABle:POWer:OUTPut:MIN:MINimum? ALL

FETCh:STABle:POWer:OUTPut:MIN:MINimum? CURRent

FETCh:STABle:POWer:OUTPut:MIN:SELected[:RESult?]

FETCh:STABle:POWer:OUTPut:MIN:STDeviation? ALL

FETCh:STABle:POWer:OUTPut:MIN:STDeviation? CURRent

Quadrature Error

FETCh:STABle:QERRor:AVERage? ALL

FETCh:STABle:QERRor:AVERage? CURRent

FETCh:STABle:QERRor:MAXimum? ALL

FETCh:STABle:QERRor:MAXimum? CURRent

FETCh:STABle:QERRor:MINimum? ALL

FETCh:STABle:QERRor:MINimum? CURRent

FETCh:STABle:QERRor:SELected[:RESult?]

FETCh:STABle:QERRor:STDeviation? ALL

FETCh:STABle:QERRor:STDeviation? CURRent

Baseband Q Input Voltage

FETCh:STABle:QVOLtage:AVG:AVERage? ALL

FETCh:STABle:QVOLtage:AVG:AVERage? CURRent

FETCh:STABle:QVOLtage:AVG:MAXimum? ALL

FETCh:STABle:QVOLtage:AVG:MAXimum? CURRent

FETCh:STABle:QVOLtage:AVG:MINimum? ALL

FETCh:STABle:QVOLtage:AVG:MINimum? CURRent

FETCh:STABle:QVOLtage:AVG:SELected[:RESult?]

FETCh:STABle:QVOLtage:AVG:STDeviation? ALL

FETCh:STABle:QVOLtage:AVG:STDeviation? CURRent

FETCh:STABle:QVOLtage:MAX:AVERage? ALL

FETCh:STABle:QVOLtage:MAX:AVERage? CURRent

FETCh:STABle:QVOLtage:MAX:MAXimum? ALL

FETCh:STABle:QVOLtage:MAX:MAXimum? CURRent

FETCh:STABle:QVOLtage:MAX:MINimum? ALL

FETCh:STABle:QVOLtage:MAX:MINimum? CURRent

FETCh:STABle:QVOLtage:MAX:SELected[:RESult?]

FETCh:STABle:QVOLtage:MAX:STDeviation? ALL

FETCh:STABle:QVOLtage:MAX:STDeviation? CURRent

FETCh:STABle:QVOLtage:MIN:AVERage? ALL

FETCh:STABle:QVOLtage:MIN:AVERage? CURRent

FETCh:STABle:QVOLtage:MIN:MAXimum? ALL

FETCh:STABle:QVOLtage:MIN:MAXimum? CURRent

FETCh:STABle:QVOLtage:MIN:MINimum? ALL

FETCh:STABle:QVOLtage:MIN:MINimum? CURRent

FETCh:STABle:QVOLtage:MIN:SELected[:RESult?]

FETCh:STABle:QVOLtage:MIN:STDeviation? ALL

FETCh:STABle:QVOLtage:MIN:STDeviation? CURRent

Raw EVM

FETCh:STABle:REVM:AVG:AVERage? ALL

FETCh:STABle:REVM:AVG:AVERage? CURRent

FETCh:STABle:REVM:AVG:MAXimum? ALL

FETCh:STABle:REVM:AVG:MAXimum? CURRent

FETCh:STABle:REVM:AVG:MINimum? ALL

FETCh:STABle:REVM:AVG:MINimum? CURRent

FETCh:STABle:REVM:AVG:SELected[:RESult?]

FETCh:STABle:REVM:AVG:STDeviation? ALL

FETCh:STABle:REVM:AVG:STDeviation? CURRent

FETCh:STABle:REVM:MAX:AVERage? ALL

FETCh:STABle:REVM:MAX:AVERage? CURRent

FETCh:STABle:REVM:MAX:MAXimum? ALL

FETCh:STABle:REVM:MAX:MAXimum? CURRent

FETCh:STABle:REVM:MAX:MINimum? ALL

FETCh:STABle:REVM:MAX:MINimum? CURRent

FETCh:STABle:REVM:MAX:SELected[:RESult?]

FETCh:STABle:REVM:MAX:STDeviation? ALL

FETCh:STABle:REVM:MAX:STDeviation? CURRent

FETCh:STABle:REVM:MIN:AVERage? ALL

FETCh:STABle:REVM:MIN:AVERage? CURRent

FETCh:STABle:REVM:MIN:MAXimum? ALL

FETCh:STABle:REVM:MIN:MAXimum? CURRent

FETCh:STABle:REVM:MIN:MINimum? ALL

FETCh:STABle:REVM:MIN:MINimum? CURRent

FETCh:STABle:REVM:MIN:SELected[:RESult?]

FETCh:STABle:REVM:MIN:STDeviation? ALL

FETCh:STABle:REVM:MIN:STDeviation? CURRent

Raw Model EVM

FETCh:STABle:RMEV:AVG:AVERage? ALL

FETCh:STABle:RMEV:AVG:AVERage? CURRent

FETCh:STABle:RMEV:AVG:MAXimum? ALL

FETCh:STABle:RMEV:AVG:MAXimum? CURRent

FETCh:STABle:RMEV:AVG:MINimum? ALL

FETCh:STABle:RMEV:AVG:MINimum? CURRent

FETCh:STABle:RMEV:AVG:SELected[:RESult?]

FETCh:STABle:RMEV:AVG:STDeviation? ALL

FETCh:STABle:RMEV:AVG:STDeviation? CURRent

FETCh:STABle:RMEV:MAX:AVERage? ALL

FETCh:STABle:RMEV:MAX:AVERage? CURRent

FETCh:STABle:RMEV:MAX:MAXimum? ALL

FETCh:STABle:RMEV:MAX:MAXimum? CURRent

FETCh:STABle:RMEV:MAX:MINimum? ALL

FETCh:STABle:RMEV:MAX:MINimum? CURRent

FETCh:STABle:RMEV:MAX:SELected[:RESult?]

FETCh:STABle:RMEV:MAX:STDeviation? ALL

FETCh:STABle:RMEV:MAX:STDeviation? CURRent

FETCh:STABle:RMEV:MIN:AVERage? ALL

FETCh:STABle:RMEV:MIN:AVERage? CURRent

FETCh:STABle:RMEV:MIN:MAXimum? ALL

FETCh:STABle:RMEV:MIN:MAXimum? CURRent

FETCh:STABle:RMEV:MIN:MINimum? ALL

FETCh:STABle:RMEV:MIN:MINimum? CURRent

FETCh:STABle:RMEV:MIN:SELected[:RESult?]

FETCh:STABle:RMEV:MIN:STDeviation? ALL

FETCh:STABle:RMEV:MIN:STDeviation? CURRent

Sample Rate Error

FETCh:STABle:SRERor:AVERage? ALL

FETCh:STABle:SRERor:AVERage? CURRent

FETCh:STABle:SRERor:MAXimum? ALL

FETCh:STABle:SRERor:MAXimum? CURRent

FETCh:STABle:SRERor:MINimum? ALL

FETCh:STABle:SRERor:MINimum? CURRent

FETCh:STABle:SRERor:SELected[:RESult?]

FETCh:STABle:SRERor:STDeviation? ALL

FETCh:STABle:SRERor:STDeviation? CURRent

Voltage (Vcc)

FETCh:STABle:VCC:AVG:AVERage? ALL

FETCh:STABle:VCC:AVG:AVERage? CURRent

FETCh:STABle:VCC:AVG:MAXimum? ALL

FETCh:STABle:VCC:AVG:MAXimum? CURRent

FETCh:STABle:VCC:AVG:MINimum? ALL

FETCh:STABle:VCC:AVG:MINimum? CURRent

FETCh:STABle:VCC:AVG:SELected[:RESult?]

FETCh:STABle:VCC:AVG:STDeviation? ALL

FETCh:STABle:VCC:AVG:STDeviation? CURRent

FETCh:STABle:VCC:MAX:AVERage? ALL

FETCh:STABle:VCC:MAX:AVERage? CURRent

FETCh:STABle:VCC:MAX:MAXimum? ALL

FETCh:STABle:VCC:MAX:MAXimum? CURRent

FETCh:STABle:VCC:MAX:MINimum? ALL

FETCh:STABle:VCC:MAX:MINimum? CURRent

FETCh:STABle:VCC:MAX:SELected[:RESult?]

FETCh:STABle:VCC:MAX:STDeviation? ALL

FETCh:STABle:VCC:MAX:STDeviation? CURRent

FETCh:STABle:VCC:MIN:AVERage? ALL

FETCh:STABle:VCC:MIN:AVERage? CURRent

FETCh:STABle:VCC:MIN:MAXimum? ALL

FETCh:STABle:VCC:MIN:MAXimum? CURRent

FETCh:STABle:VCC:MIN:MINimum? ALL

FETCh:STABle:VCC:MIN:MINimum? CURRent

FETCh:STABle:VCC:MIN:SELected[:RESult?]

FETCh:STABle:VCC:MIN:STDeviation? ALL

FETCh:STABle:VCC:MIN:STDeviation? CURRent

6.5.4 Retrieving I/Q Data

TRACe:IQ:EQUalized? <Input>

This command queries the equalized I/Q data.

Prerequisites for this command

- Equalized data must be available.

Query parameters:

<Input> RF

You have to state this parameter, but it is always "RF".

Return values:

<Result> String containing the I/Q data.

Example: //Query equalized I/Q data
`TRAC:IQ:EQU? RF`
//Query raw I/Q data
`TRAC:IQ:DATA?`

Usage: Query only

Manual operation: See "Using the equalizer" on page 75

TRACe:IQ:SYNChronized? <InpMode>

This command queries the (measured) synchronized I/Q data (which corresponds to the green bar in the Magnitude Capture result display).

Query parameters:

<InpMode> **RF**
Queries the data captured on the RF input.

Return values:

<Result> String containing the synchronized measurement values.

Example: `TRAC:IQ:SYNC? RF`
would return, e.g.
`'-40.376233,-39.982912,...'`

Usage: Query only

6.6 Configuring Amplifier Measurements

6.6.1 Designing a Reference Signal

CONFigure:REFSignal:CGW:AMODe[:STATe] <State>

Sets and queries the "Force ARB Mode" setting.

Parameters:

<State> ON | OFF | 1 | 0

Example: `CONFigure:REFSignal:CGW:AMODe:STAT ON`

Manual operation: See "Designing a reference signal on a signal generator" on page 37

CONFigure:REFSignal:CGW:LEDState?

This command queries the processing state of the reference signal generation if the reference signal was designed on a signal generator.

Prerequisites for this command

- Configure reference signal on a signal generator.

Return values:

<State>

GREen
Reference signal was successfully generated and loaded into the application.

GREY
Unknown processing state.

RED
Reference signal was not successfully generated or loaded into the application.

Example:
```
CONF:REFS:CGW:READ
CONF:REFS:CGW:LEDS?
```
would return, e.g.
```
GRE
```

Usage: Query only

Manual operation: See "Designing a reference signal on a signal generator" on page 37

CONFigure:REFSignal:CGW:READ

This command transfers a reference signal designed on a signal generator into the R&S FPS-K18.

Example: //Import reference signal data from the generator
```
CONF:REFS:CGW:READ
```

Usage: Event

Manual operation: See "Designing a reference signal on a signal generator" on page 37

CONFigure:REFSignal:CWF:DPIPower <Power>

This command defines the peak input power of the DUT.

This is necessary when you turn off `CONFigure:REFSignal:CWF:ETGenerator[:STATe]` (otherwise, the command has no effect).

Prerequisites for this command

- Generate reference signal with a waveform file

Parameters:

<Power> <numeric value>

Default unit: dBm

Example: //Define DUT input power

```
CONF:REFS:CWF:ETG OFF
CONF:REFS:CWF:DPIP 3
```

Manual operation: See "Designing a reference signal in a waveform file" on page 38

CONFigure:REFSignal:CWF:ETGenerator[:STATe] <State>

This command turns the transfer of the reference signal data to a generator on and off.

Prerequisites for this command

- Generate reference signal with a waveform file

Parameters:

<State>

ON | 1

Reference signal data is transferred to the generator and generated with the generator.

OFF | 0

Reference signal data is loaded into the application without transferring the waveform to the generator.

When you turn it off, you have to define the peak input power of the DUT with `CONFigure:REFSignal:CWF:DPIPower`. Otherwise, measurement result may be invalid.

*RST: 1

Example: //Generate reference signal without transferring the waveform file to a generator.

```
CONF:REFS:CWF:ETG OFF
```

Manual operation: See "Designing a reference signal in a waveform file" on page 38

CONFigure:REFSignal:CWF:FPATh <FileName>

This command selects a waveform file containing a reference signal.

Parameters:

<FileName> String containing the name and path to the waveform file.

Example: //Select a waveform file
`CONF:REFS:CWF:FPAT 'C:\RefSignal.wv'`

Manual operation: See "Designing a reference signal in a waveform file" on page 38

CONFigure:REFSignal:CWF:LEDState?

This command queries the processing status of a reference signal generated with a waveform file.

Available when you generate the reference signal with a waveform file.

Return values:

<State>

GREen
The reference signal was successfully loaded into the application.
When `CONFigure:REFSignal:CWF:ETGenerator[:STATe]` = ON, this also indicates that the waveform file was accepted by the signal generator.

GREY
Unknown processing state.

RED
The reference signal could not have been loaded into the application.
When `CONFigure:REFSignal:CWF:ETGenerator[:STATe]` = ON, this could also mean that the waveform file was not accepted by the signal generator.

Example:
`CONF:REFS:CWF:FPAT 'C:\RefSignal.wv'`
`CONF:REFS:CWF:WRITE`
`CONF:REFS:CWF:LEDS?`
would return, e.g.
`GRE`

Usage: Query only

Manual operation: See "Designing a reference signal in a waveform file" on page 38

CONFigure:REFSignal:CWF:WRITe

This command loads a reference signal based on a waveform file into the application.

When you turn on the reference signal export to the generator (CONFigure:REFSignal:CWF:ETGenerator[:STATe]), the command also transfers the waveform file to the generator.

Make sure to synchronize with *OPC? or *WAI to make sure that the command was successfully applied on the generator before sending the next command.

Example: //Load the reference signal into the application and, if the feature has been turned on, transfer the reference signal to the generator

```
CONF:REFS:CWF:FPAT 'C:\RefSignal.wv';
CONF:REFS:CWF:WRITE;*WAI
```

Usage: Event

Manual operation: See "Designing a reference signal in a waveform file" on page 38

CONFigure:REFSignal:GOS:BWIDth <Bandwidth>

This command defines the bandwidth of the internally generated reference signal.

Parameters:

<Bandwidth> <numeric value>

Default unit: Hz

Example: //Define reference signal bandwidth

```
CONF:REFS:GOS:BWID 10MHZ
```

Manual operation: See "Signal Bandwidth" on page 41

CONFigure:REFSignal:GOS:CRESt <CrestFactor>

This command defines the crest factor of the internally generated reference signal.

Parameters:

<CrestFactor> <numeric value>

Default unit: dB

Example: //Define crest factor

```
CONF:REFS:GOS:CRES 15
```

Manual operation: See "Crest Factor" on page 42

CONFigure:REFSignal:GOS:DCYCle <DutyCycle>

This command defines the duty cycle of an internally generated pulsed reference signal.

Parameters:

<DutyCycle> <numeric value>

Default unit: %

Example: //Define duty cycle
`CONF:REFS:GOS:DCYC 75`

Manual operation: See "Pulse Duty Cycle" on page 42

CONFigure:REFSignal:GOS:LEDState?

This command queries the processing status of an internally generated reference signal.

Available when you configure the reference signal within the R&S FPS-K18.

Return values:

<State>

GREen
Generation of the internally generated reference signal was successful. Transmission of the waveform file to the signal generator was also successful.

GREY
Unknown transmission state.

RED
Generation and / or transmission of the internally generated reference signal was not successful.

Example:
`CONF:REFS:GOS:WRITE`
`CONF:REFS:GOS:LEDS?`
would return, e.g.
`GRE`

Usage: Query only

Manual operation: See "Designing a reference signal within the R&S FPS-K18" on page 40

CONFigure:REFSignal:GOS:NPOSition <Frequency>

This command defines the offset of a notch relative to the center frequency in the internally generated reference signal.

Parameters:

<Frequency> <numeric value>

Default unit: Hz

Example: //Define a notch offset
`CONF:REFS:GOS:NPOS 10000`

Manual operation: See "Notch Position" on page 43

CONFigure:REFSignal:GOS:NWIDth <Frequency>

This command defines the notch width of an internally generated reference signal.

Parameters:

<Frequency> <numeric value>

Default unit: Hz

Example: //Define notch width
`CONF:REFS:GOS:NWID 150000`

Manual operation: See "Notch Width" on page 42

CONFigure:REFSignal:GOS:RLENgth <Samples>

This command defines the ramp length of an internally generated pulsed reference signal.

Parameters:

<Samples> <numeric value>: (integer only)

Number of samples on each side of the pulse (= ramp length).

Default unit: Samples

Example: //Define ramp length
`CONF:REFS:GOS:RLEN 5`

Manual operation: See "Ramp Length" on page 42

CONFigure:REFSignal:GOS:SLENgth <Samples>

This command defines the length of the internally generated reference signal.

Parameters:

<Samples> <numeric value>: (integer only)

Default unit: Samples

Example: //Define reference signal size
`CONF:REFS:GOS:SLEN 1024`

Manual operation: See "Signal Length" on page 42

CONFigure:REFSignal:GOS:SRATe <SampleRate>

This command defines the clock (or sample) rate of the internally generated reference signal.

Parameters:

<SampleRate> <numeric value>

Default unit: Hz

Example: //Defines sample rate
`CONF:REFS:GOS:SRAT 20000000`

CONFigure:REFSignal:GOS:WNAMe <FileName>

This command defines a file name for the waveform of the reference signal.

Parameters:

<FileName> String containing the name of the waveform file.
The file extension (.wv) is added automatically.

Example: //Define name for the waveform file
`CONF:REFS:GOS:WNAM 'RefSignal'`

Manual operation: See "Waveform File Name" on page 42

CONFigure:REFSignal:GOS:WRITe

This command internally generates the reference signal based on the signal characteristics that you have defined.

The waveform file that has been created is loaded into the DSP of the R&S FPS-K18 and is additionally transferred into the ARB of the signal generator.

Make sure to synchronize with `*OPC?` or `*WAI` to make sure that the command was successfully applied on the generator before sending the next command.

Example: //Generate the reference signal and transfer it into the R&S FPS-K18. In addition, the waveform file that has been created is transferred into the signal generator.
`CONF:REFS:GOS:WRIT;*WAI`

Usage: Event

Manual operation: See "Designing a reference signal within the R&S FPS-K18" on page 40

CONFigure:REFSignal:SEGMent <Segment>

This command selects the segment of the reference signal that should be used in the measurement when the reference signal is based on a multi segment waveform file.

Parameters:

<Segment> <numeric value>: (integer only)

Range: Depends on the number of segments in the waveform file.

*RST: 0

Example: //Select a segment
`CONF:REFS:SEGM 3`

Manual operation: See "Using multi-segment waveform files" on page 36

CONFigure:REFSignal:SINFo:FPATh?

This command queries the file name and location of the currently used reference signal.

Return values:

<FileName> String containing the file name and location of the file.

Example: `CONF:REFS:SINF:FPAT?`
would return, e.g.
`C:\waveform.wv`

Usage: Query only

Manual operation: See "Reference signal information" on page 36

CONFigure:REFSignal:SINFo:SLENgth?

This command queries the sample length of the currently used reference signal.

Return values:
<Samples> <numeric value>: (integer only)
Default unit: Samples

Example: `CONF:REFS:SINF:SLEN?`
would return, e.g.
`40000`

Usage: Query only

Manual operation: See "Reference signal information" on page 36

CONFigure:REFSignal:SINFo:SRATe?

This command queries the sample rate of the currently used reference signal.

Return values:
<SampleRate> <numeric value>
Default unit: Hz

Example: `CONF:REFS:SINF:SRAT?`
would return, e.g.
`32000000`

Usage: Query only

Manual operation: See "Reference signal information" on page 36

CONFigure:REFSignal:SINFo:CFACtor?

Returns the crest factor of the reference signal.

Return values:
<CrestFactor> <numeric value>

Example: `CONFigure:REFSignal:SINFo:CFACtor?`

Usage: Query only

Manual operation: See "Reference signal information" on page 36

CONFigure:REFSignal:SINFo:OBW?

Returns the occupied bandwidth of the reference signal.

Return values:
<Bandwidth> <numeric value>

Example: `CONFigure:REFSignal:SINFo:OBW?`

Usage: Query only

Manual operation: See "Reference signal information" on page 36

CONFigure:CFReduction[:STATe] <State>

Enables the crest factor reduction calculation.

Parameters:
<State> 0 | 1 | OFF | ON
*RST: 0

Example: `CONFigure:CFR ON`

CONFigure:CFReduction[:STATe]:LEDState?

Reads the LED status of the crest factor reduction calculation.

Parameters:
<State> GREY | RED | GREen

Example: `CONFigure:CFReduction:STATe:LEDState`

Usage: Query only

CONFigure:CFReduction:SBANdwidth <Bandwidth>

Sets and queries the signal bandwidth.

Parameters:
<Bandwidth> <numeric value>
Default unit: Hz

Example: `CONF:CFR:SBAN 10MHz`

CONFigure:CFReduction:SBANdwidth:AUTO <State>

Sets and queries the signal bandwidth mode.

Parameters:
<State> 0 | 1 | OFF | ON

Example: `CONFigure:CFReduction:SBANdwidth:AUTO ON`

CONFigure:CFReduction:SBANdwidth:LEDState?

Reads the LED status of the signal bandwith.

Parameters:

<State> GREY | RED | GREen

Example: `CONFigure:CFReduction:SBANdwidth:LEDState`

Usage: Query only

CONFigure:CFReduction:RSORignal <State>

Switches the EVM reference signal.

Parameters:

<State> 0 | 1 | OFF | ON

Example: `CONFigure:CFReduction:RSORignal ON`

CONFigure:CFReduction:ITERations <Iterations>

Sets and queries the crest factor reduction maximum iterations.

Parameters:

<Iterations> <numeric value>

Example: `CONFigure:CFReduction:ITERations 2`

CONFigure:CFReduction:ITERations:LEDState?

Reads the LED status of the crest factor reduction maximum iterations.

Parameters:

<State> GREY | RED | GREen

Example: `CONFigure:CFReduction:ITERations:LEDState`

Usage: Query only

CONFigure:CFReduction:FILTer <FilterMode>

Selects simple or enhanced filter mode for crest factor reduction.

Parameters:

<FilterMode> SIMPle | ENHanced

Example: `CONFigure:CFReduction:FILTer ENH`

CONFigure:CFReduction:FILTer:LEDState?

Reads the LED status of crest factor reduction filter mode.

Return values:

<State> GREY | RED | GREEn

Example: `CONFigure:CFReduction:FILTer:LEDState`

Usage: Query only

CONFigure:CFReduction:CSPacing <Spacing>

Sets and queries the crest factor reduction channel spacing.

Parameters:

<Spacing> <numeric value>

Default unit: Hz

Example: `CONF:CFR:CSP 10MHz`

CONFigure:CFReduction:CSPacing:AUTO <State>

Sets and queries the crest factor reduction channel spacing mode.

Parameters:

<State> 0 | 1 | OFF | ON

Example: `CONFigure:CFReduction:CSPacing:AUTO ON`

CONFigure:CFReduction:CSPacing:LEDState?

Reads the LED status of the crest factor reduction channel spacing.

Parameters:

<State> GREY | RED | GREEn

Example: `CONFigure:CFReduction:CSPacing:LEDState`

Usage: Query only

CONFigure:CFReduction:CFDelta <CFDelta>

Sets the value difference by which you want to change the crest factor.

Parameters:

<CFDelta> <numeric value>

Default unit: dB

Example: `CONFigure:CFReduction:CFDelta 10`

CONFigure:CFReduction:CFDelta:LEDState?

Reads the LED status of the crest factor delta.

Parameters:

<State> GREY | RED | GREEn

Example: CONFigure:CFReduction:CFDelta:LEDState

Usage: Query only

CONFigure:CFReduction:CCFactor?

Queries the crest factor of the waveform after the calculation of the resulting crest factor is completed.

Return values:
<CCF>

Example: CONFigure:CFReduction:CCFactor?

Usage: Query only

CONFigure:CFReduction:APPLy

Applies crest factor reduction on the connected signal generator.

Only available for backward compatibility, use CONFigure:CFReduction:READ on page 221 instead.

Example: CONFigure:CFReduction:APPLy

Usage: Event

CONFigure:CFReduction:APPLy:LEDState?

Reads the LED status of crest factor reduction apply on the connected signal generator.

Only available for backward compatibility, use CONFigure:CFReduction:READ:LEDState? on page 221 instead.

Return values:
<State> GREY | RED | GREen

Example: CONFigure:CFReduction:APPLy:LEDState

Usage: Query only

CONFigure:CFReduction:READ

Applies crest factor reduction on the connected signal generator.

Example: CONFigure:CFReduction:READ

Usage: Event

CONFigure:CFReduction:READ:LEDState?

Reads the LED status of crest factor reduction apply on the connected signal generator.

Return values:
<State> GREY | RED | GREen

Example: `CONFigure:CFReduction:READ:LEDState`

Usage: Query only

CONFigure:CFReduction:MFORder <MaximumFilterOrder>

Sets and queries the maximum filter order for crest factor reduction.

Parameters:
<MaximumFilterOrder> numeric value

Example: `CONF:CFR:MFOR 100`

CONFigure:CFReduction:MFORder:LEDState?

Reads the LED status of crest factor reduction maximum filter order.

Return values:
<State> GREY | RED | GREen

Example: `CONFigure:CFReduction:MFORder:LEDState`

Usage: Query only

CONFigure:CFReduction:PFRequency <Time>

Sets and queries the passband frequency for crest factor reduction.

Parameters:
<Time> numeric value
Default unit: Hz

Example: `CONF:CFR:PFR 10MHz`

CONFigure:CFReduction:PFRequency:LEDState?

Reads the LED status of crest factor reduction passband frequency.

Return values:
<State> GREY | RED | GREen

Example: `CONFigure:CFReduction:PFRequency:LEDState`

Usage: Query only

CONFigure:CFReduction:SFRequency <Time>

Sets and queries the stopband frequency for crest factor reduction.

Parameters:

<Time>	numeric value Default unit: Hz
Example:	`CONF:CFR:SFR 10MHz`

CONFigure:CFReduction:SFRequency:LEDState?

Reads the LED status of crest factor reduction stopband frequency.

Return values:

<State>	GREY | RED | GREen
Example:	`CONFigure:CFReduction:SFRequency:LEDState`
Usage:	Query only

6.6.2 Selecting and Configuring the Input Source

INPut<ip>:COUPling <CouplingType>

This command selects the coupling type of the RF input.

Suffix:

<ip>	1 | 2 irrelevant

Parameters:

<CouplingType>	AC | DC **AC** AC coupling **DC** DC coupling *RST: AC
Example:	`INP:COUP DC`
Manual operation:	See " Input Coupling " on page 44

INPut<ip>:DPATh <DirectPath>

Enables or disables the use of the direct path for frequencies close to 0 Hz.

Suffix:
<ip> 1 | 2
irrelevant

Parameters:
<DirectPath> AUTO | OFF

AUTO | 1
(Default) the direct path is used automatically for frequencies close to 0 Hz.

OFF | 0
The analog mixer path is always used.

Example: `INP:DPAT OFF`

INPut<ip>:FILE:PATH <FileName>[, <AnalysisBW>]

This command selects the I/Q data file to be used as input for further measurements.

The I/Q data must have a specific format as described in R&S FPS I/Q Analyzer and I/Q Input User Manual.

Suffix:
<ip> 1 | 2
irrelevant

Parameters:
<FileName> String containing the path and name of the source file.
The file extension is `*.iq.tar`.

<AnalysisBW> Optionally: The analysis bandwidth to be used by the measurement. The bandwidth must be smaller than or equal to the bandwidth of the data that was stored in the file.
Default unit: HZ

Example: `INP:FILE:PATH 'C:\R_S\Instr\user\data.iq.tar'`
Uses I/Q data from the specified file as input.

Manual operation: See " Select I/Q data file " on page 45

INPut<ip>:FILTer:YIG[:STATe] <State>

Enables or disables the YIG filter.

Suffix:
<ip> 1 | 2
irrelevant

Parameters:
<State> ON | OFF | 0 | 1

Example: `INP:FILT:YIG OFF`
Deactivates the YIG-preselector.

Manual operation: See " YIG-Preselector " on page 44

INPut<ip>:IMPedance <Impedance>

This command selects the nominal input impedance of the RF input. In some applications, only 50 Ω are supported.

Suffix:
<ip> 1 | 2
irrelevant

Parameters:
<Impedance> 50 | 75
*RST: 50 Ω
Default unit: OHM

Example: INP:IMP 75

Manual operation: See " Impedance " on page 44

INPut<ip>:SELect <Source>

This command selects the signal source for measurements, i.e. it defines which connector is used to input data to the R&S FPS.

Suffix:
<ip> 1 | 2
irrelevant

Parameters:
<Source> **RF**
Radio Frequency ("RF INPUT" connector)
FIQ
I/Q data file
(selected by INPut<ip>:FILE:PATH on page 224)
*RST: RF

Manual operation: See "I/Q Input File State" on page 45

6.6.3 Configuring the Frequency

[SENSe:]FREQuency:CENTer <Frequency>

This command defines the center frequency.

Parameters:
<Frequency> The allowed range and f_{max} is specified in the data sheet.
*RST: fmax/2
Default unit: Hz

Example:
`FREQ:CENT 100 MHz`
`FREQ:CENT:STEP 10 MHz`
`FREQ:CENT UP`
Sets the center frequency to 110 MHz.

Manual operation: See "Center Frequency" on page 46

[SENSe:]FREQuency:CENTer:STEP <StepSize>

This command defines the center frequency step size.

Parameters:
<StepSize> f_{max} is specified in the data sheet.

Range: 1 to fMAX
*RST: 0.1 x span
Default unit: Hz

Example:
//Set the center frequency to 110 MHz.
`FREQ:CENT 100 MHz`
`FREQ:CENT:STEP 10 MHz`
`FREQ:CENT UP`

Manual operation: See "Center Frequency Stepsize" on page 46

[SENSe:]FREQuency:OFFSet <Offset>

This command defines a frequency offset.

If this value is not 0 Hz, the application assumes that the input signal was frequency shifted outside the application. All results of type "frequency" will be corrected for this shift numerically by the application.

Parameters:
<Offset> Range: -1 THz to 1 THz
*RST: 0 Hz
Default unit: HZ

Example: `FREQ:OFFS 1GHZ`

Manual operation: See " Frequency Offset " on page 47

6.6.4 Defining Level Characteristics

DISPlay[:WINDow<n>]:TRACe<t>:Y[:SCALe]:RLEVel <ReferenceLevel>

This command defines the reference level (for all traces in all windows).

Suffix:

<n> irrelevant

<t> irrelevant

Parameters:

<ReferenceLevel> The unit is variable.

Range: see datasheet
*RST: 0 dBm
Default unit: DBM

Example: `DISP:TRAC:Y:RLEV -60dBm`

Manual operation: See " Reference Level " on page 48

DISPlay[:WINDow<n>]:TRACe<t>:Y[:SCALe]:RLEVel:OFFSet <Offset>

This command defines a reference level offset (for all traces in all windows).

Suffix:

<n> irrelevant

<t> irrelevant

Parameters:

<Offset> Range: -200 dB to 200 dB
*RST: 0dB
Default unit: DB

Example: `DISP:TRAC:Y:RLEV:OFFS -10dB`

Manual operation: See " Shifting the Display (Offset)" on page 48

INPut<ip>:ATTenuation <Attenuation>

This command defines the total attenuation for RF input.

If you set the attenuation manually, it is no longer coupled to the reference level, but the reference level is coupled to the attenuation. Thus, if the current reference level is not compatible with an attenuation that has been set manually, the command also adjusts the reference level.

Suffix:

<ip> 1 | 2
irrelevant

Parameters:

<Attenuation> Range: see data sheet
Increment: 5 dB (with optional electr. attenuator: 1 dB)
*RST: 10 dB (AUTO is set to ON)
Default unit: DB

Example: `INP:ATT 30dB`
Defines a 30 dB attenuation and decouples the attenuation from the reference level.

Manual operation: See " Attenuation Mode / Value " on page 49

INPut<ip>:ATTenuation:AUTO <State>

This command couples or decouples the attenuation to the reference level. Thus, when the reference level is changed, the R&S FPS determines the signal level for optimal internal data processing and sets the required attenuation accordingly.

Suffix:
<ip> — 1 | 2
irrelevant

Parameters:
<State> — ON | OFF | 0 | 1
*RST: 1

Example: `INP:ATT:AUTO ON`
Couples the attenuation to the reference level.

Manual operation: See " Attenuation Mode / Value " on page 49

INPut<ip>:EATT <Attenuation>

This command defines an electronic attenuation manually. Automatic mode must be switched off (`INP:EATT:AUTO OFF`, see `INPut<ip>:EATT:AUTO` on page 228).

If the current reference level is not compatible with an attenuation that has been set manually, the command also adjusts the reference level.

Suffix:
<ip> — 1 | 2
irrelevant

Parameters:
<Attenuation> — attenuation in dB

Range: see data sheet
Increment: 1 dB
*RST: 0 dB (OFF)
Default unit: DB

Example:
```
INP:EATT:AUTO OFF
INP:EATT 10 dB
```

Manual operation: See " Using Electronic Attenuation " on page 49

INPut<ip>:EATT:AUTO <State>

This command turns automatic selection of the electronic attenuation on and off.

If on, electronic attenuation reduces the mechanical attenuation whenever possible.

Suffix:
<ip> 1 | 2
irrelevant

Parameters:
<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

*RST: 1

Example: `INP:EATT:AUTO OFF`

Manual operation: See " Using Electronic Attenuation " on page 49

INPut<ip>:EATT:STATe <State>

This command turns the electronic attenuator on and off.

Suffix:
<ip> 1 | 2
irrelevant

Parameters:
<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

*RST: 0

Example: `INP:EATT:STAT ON`
Switches the electronic attenuator into the signal path.

Manual operation: See " Using Electronic Attenuation " on page 49

INPut<ip>:GAIN:STATe <State>

This command turns the internal preamplifier on and off. It requires the optional preamplifier hardware.

If activated, the input signal is amplified by 20 dB.

If option R&S FPS-B22 is installed, the preamplifier is only active below 7 GHz.

If option R&S FPS-B24 is installed, the preamplifier is active for all frequencies.

Suffix:
<ip> 1 | 2
irrelevant

Parameters:
<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

*RST: 0

Example: `INP:GAIN:STAT ON`
Switches on 20 dB preamplification.

Manual operation: See " Preamplifier (option B22/B24)" on page 48

6.6.5 Controlling a Signal Generator

When you configure the signal generator, make sure to synchronize with `*OPC?` or `*WAI` to make sure that the command was successfully applied on the generator before sending the next command.

CONFigure:DUT:STIMe <Time>

This command defines the settling time between generator setting changes and the start of the next measurement.

Parameters:

<Time> <numeric value>

*RST: 0

Default unit: s

Example: //Define settling delay

`CONF:DUT:STIM 0.5`

Manual operation: See "Settling Delay" on page 55

CONFigure:GENerator:CONTrol[:STATe] <State>

This command turns the generator control on and off.

Parameters:

<State> ON | OFF | 1 | 0

*RST: OFF

Example: //Turn on generator control

`CONF:GEN:CONT ON`

Manual operation: See "Generator Control State" on page 52

CONFigure:GENerator:DUT:INPut:MAXimum:POWer <Level>

This command defines the maximum generator output power.

Parameters:

<Level> Default unit: dBm

Example: //Define maximum output power

`CONF:GEN:DUT:INP:MAX:POW 0DBM`

Manual operation: See "Maximum DUT Input Level" on page 53

CONFigure:GENerator:DUT:INPut:MAXimum:POWer:LEDState?

This command queries the maximum output level configuration state on the generator.

Return values:

<State> **GREen**

Configuration was successful.

GREY

Unknown configuration state.

RED

Configuration to the reference was not successful.

Example: `CONF:GEN:DUT:INP:MAX:POW:LEDS?`
would return, e.g.:
`GRE`

Usage: Query only

Manual operation: See "Maximum DUT Input Level" on page 53

CONFigure:GENerator:EXTernal:ROSCillator <Source>

This command selects the source of the generator reference frequency.

Make sure to synchronize with `*OPC?` or `*WAI` to make sure that the command was successfully applied on the generator before sending the next command.

Parameters:

<Source> **EXT**
The generator uses an external reference frequency (for example that of the R&S FPS).

INT
The generator uses its own (internal) reference frequency.

Example: //Select the reference frequency of the generator
`CONF:GEN:EXT:ROSC INT;*WAI`

Manual operation: See "Reference Frequency" on page 54

CONFigure:GENerator:EXTernal:ROSCillator:LEDState?

This command queries the connection status of the generator to its frequency reference.

Return values:

<State> **GREen**
Connection to the reference was successful.

GREY
Unknown connection state.

RED
Connection to the reference was not successful.

Example: `CONF:GEN:EXT:ROSC:LEDS?`
would return, e.g.:
`RED`

Usage: Query only

Manual operation: See "Reference Frequency" on page 54

CONFigure:GENerator:FREQuency:CENTer <Frequency>

This command defines the frequency of the generator.

Make sure to synchronize with `*OPC?` or `*WAI` to make sure that the command was successfully applied on the generator before sending the next command.

Parameters:

<Frequency> <numeric value>

Default unit: Hz

Example: //Define a generator frequency
`CONF:GEN:FREQ:CENT 100000000;*WAI`

Manual operation: See "Center Frequency" on page 53

CONFigure:GENerator:FREQuency:CENTer:LEDState?

This command queries the status of frequency synchronization.

Return values:

<State>

GREen
Frequency synchronization was successful.

GREY
Unknown frequency synchronization state.

RED
Frequency synchronization was not successful.

Example: `CONF:GEN:FREQ:CENT:LEDS?`
would return, e.g.:
`GRE`

Usage: Query only

Manual operation: See "Center Frequency" on page 53

CONFigure:GENerator:FREQuency:CENTer:SYNC[:STATe] <State>

This command turns synchronization of the analyzer and generator frequency on and off.

Make sure to synchronize with `*OPC?` or `*WAI` to make sure that the command was successfully applied on the generator before sending the next command.

Parameters:

<State> ON | OFF | 1 | 0

Example: //Match the generator frequency to the analyzer frequency when frequency on the R&S FPS is changed
`CONF:GEN:FREQ:CENT:SYNC ON;*WAI`

Manual operation: See "Attach to Analyzer Frequency" on page 53

CONFigure:GENerator:HSConnection:LEDState?

This command queries the state of the high speed network connection between the R&S FPS and the connected signal generator.

Return values:
<State>
GREen
High speed connection was successful.
GREY
Unknown connection state.
RED
High speed connection was not successful.

Example: `CONF:GEN:HSC:LEDS?`
would return, e.g.:
`GRE`

Usage: Query only

CONFigure:GENerator:IPConnection:ADDRess <IPAddress>

This command defines the IP address of the connected signal generator.

Make sure to synchronize with `*OPC?` or `*WAI` to make sure that the command was successfully applied on the generator before sending the next command.

Parameters:
<IPAddress> String containing the IP address or computer name.

Example: //Connect to the generator with the stated IP address
`CONF:GEN:IPC:ADDR '192.0.2.0';*WAI`

Example: //Connect to the generator with a computer name
`CONF:GEN:IPC:ADDR 'MyGenerator';*WAI`

Manual operation: See "IP Address" on page 52

CONFigure:GENerator:IPConnection:LEDState?

This command queries the state of connection to the signal generator.

Return values:
<State>
GREen
Connection was successful.
GREY
Unknown connection state.
RED
Connection was not successful.

Example: `CONF:GEN:IPC:LEDS?`
would return, e.g.:
`RED`

Usage: Query only

Manual operation: See "IP Address" on page 52

CONFigure:GENerator:POWer:LEVel <Level>

This command defines the signal generator level.

Make sure to synchronize with `*OPC?` or `*WAI` to make sure that the command was successfully applied on the generator before sending the next command.

Parameters:

<Level> <numeric value>

Default unit: dBm

Example: //Define generator output level
`CONF:GEN:POW:LEV 0;*WAI`

Manual operation: See "RMS Level" on page 52

CONFigure:GENerator:POWer:LEVel:ATTenuation <Level>

This command defines digital attenuation that is applied to digitally modulated I/Q signals.

Make sure to synchronize with `*OPC?` or `*WAI` to make sure that the command was successfully applied on the generator before sending the next command.

Parameters:

<Level> <numeric value>

*RST: 0
Default unit: dB

Example: //Attenuate the signal
`CONF:GEN:POW:LEV:ATT 10;*WAI`

Manual operation: See "Digital Attenuation" on page 54

CONFigure:GENerator:POWer:LEVel:ATTenuation:LEDState?

This command queries the configuration state of digital attenuation on the generator.

Return values:

<State>

GREen
Digital attenuation configuration was successful.

GREY
Unknown digital attenuation configuration state.

RED
Digital attenuation configuration was not successful.

Example: `CONF:GEN:POW:LEV:ATT:LEDS?`
would return, e.g.:
`RED`

Usage: Query only

CONFigure:GENerator:POWer:LEVel:LEDState?

This command queries the level configuration state on the generator.

Return values:

<State>

GREen
Level configuration was successful.

GREY
Unknown level configuration state.

RED
Level configuration was not successful.

Example:

```
CONF:GEN:POW:LEV:LEDS?
```

would return, e.g.:

```
GRE
```

Usage: Query only

Manual operation: See "RMS Level" on page 52

CONFigure:GENerator:POWer:LEVel:OFFSet <Level>

This command defines a mathematical level offset for the signal generator (for example to take external attenuation into account).

Make sure to synchronize with `*OPC?` or `*WAI` to make sure that the command was successfully applied on the generator before sending the next command.

Parameters:

<Level>

<numeric value>

Default unit: dBm

Example:

//Define a level offset

```
CONF:GEN:POW:LEV:OFFS 10;*WAI
```

Manual operation: See "RMS Level" on page 52

CONFigure:GENerator:POWer:LEVel:OFFSet:LEDState?

This command queries the level offset configuration state on the generator.

Return values:

<State>

GREen
Level offset configuration was successful.

GREY
Unknown level offset configuration state.

RED
Level offset configuration was not successful.

Example: `CONF:GEN:POW:LEV:LEDS?`
would return, e.g.:
`GRE`

Usage: Query only

Manual operation: See "RMS Level" on page 52

CONFigure:GENerator:RFOutput:LEDState?

This command queries the RF output state on the generator.

Return values:

<State>

GREen
Output configuration was successful.

GREY
Unknown output configuration state.

RED
Output configuration was not successful.

Example: `CONF:GEN:RFO:LEDS?`
would return, e.g.:
`GRE`

Usage: Query only

Manual operation: See "RF Output" on page 54

CONFigure:GENerator:RFOutput[:STATe] <State>

This command turns the RF output on the connected signal generator on and off.

Make sure to synchronize with `*OPC?` or `*WAI` to make sure that the command was successfully applied on the generator before sending the next command.

Parameters:

<State> ON | OFF | 1 | 0

*RST: 1

Example: //Turn off the RF output
`CONF:GEN:RFO OFF;*WAI`

Manual operation: See "RF Output" on page 54

CONFigure:GENerator:SEGMent <Segment>

This command selects the segment in a multi-waveform file that should be selected on the signal generator.

Make sure to synchronize with `*OPC?` or `*WAI` to make sure that the command was successfully applied on the generator before sending the next command.

Parameters:

<Segment> <numeric value>: (integer only)

Range: Depends on the number of segments in the waveform file.

*RST: 0

Example: //Select the 3rd segment of a waveform file
`CONF:GEN:SEGM 3;*WAI`

Manual operation: See "Segment" on page 54

CONFigure:GENerator:SEGMent:LEDState?

This command queries if the proper segment of a multi waveform has been selected.

Return values:

<State>

GREen
The desired segment has been selected.

GREY
Unknown segment selection state.

RED
The desired segment has not been selected.

Example: `CONF:GEN:SEGM:LEDS?`
would return, e.g.
`RED`

Usage: Query only

Manual operation: See "Segment" on page 54

CONFigure:GENerator:SETTings:UPDate

This command updates the generator settings as defined within the R&S FPS-K18.

Make sure to synchronize with `*OPC?` or `*WAI` to make sure that the command was successfully applied on the generator before sending the next command.

Example: //Update generator settings
`CONF:GEN:SETT:UPD;*WAI`

Usage: Event

CONFigure:GENerator:TARGet:PATH:BB?

This command queries the signal path of the R&S SMW used for baseband signal generation.

Note that the baseband path is always the same as the RF path selected with `CONFigure:GENerator:TARGet:PATH:RF`.

Return values:

<Path> A | B

Example: `CONF:GEN:TARG:PATH:BB?`
would return, e.g.
`A`

Usage: Query only

Manual operation: See "Path RF / BB" on page 54

CONFigure:GENerator:TARGet:PATH:RF <Path>

This command selects the signal path of the generator used for RF signal generation.

Make sure to synchronize with `*OPC?` or `*WAI` to make sure that the command was successfully applied on the generator before sending the next command.

Parameters:
<Path> A | B

Example: //Select RF path A to generate the signal
`CONF:GEN:TARG:PATH:RF A;*WAI`

Manual operation: See "Path RF / BB" on page 54

CONFigure:SETTings

This command transfers the current generator configuration into the amplifier application.

Make sure to synchronize with `*OPC?` or `*WAI` to make sure that the command was successfully applied on the generator before sending the next command.

Example: //Synchronize the generator configuration
`CONF:SETT;*WAI`

Usage: Event

6.6.6 Configuring the Data Capture

[SENSe:]BANDwidth[:RESolution] <Bandwidth>

This command defines the resolution bandwidth applied to spectrum measurements.

Prerequisites for this command

- Turn off automatic selection of RBW (`[SENSe:]BANDwidth[:RESolution]:AUTO`).

Parameters:

<Bandwidth> <numeric value>

Range: 1 Hz to 10 MHz
*RST: RBW is selected automatically
Default unit: Hz

Example: //Select resolution bandwidth
```
BAND:AUTO OFF
BAND 100KHZ
```

Manual operation: See "Defining the resolution bandwidth for spectrum measurements" on page 68

[SENSe:]BANDwidth[:RESolution]:AUTO <State>

This command turns automatic selection of the resolution bandwidth (RBW) for spectrum measurements on and off.

Parameters:

<State> ON | OFF | 1 | 0

*RST: 1

Example: //Select manual resolution bandwidth
```
BAND:AUTO OFF
BAND 100KHZ
```

Manual operation: See "Defining the resolution bandwidth for spectrum measurements" on page 68

[SENSe:]REFSig:TIME?

This command queries the length of the reference signal as shown in the "Acquisition" dialog box.

Return values:

<Duration> <numeric value>

Default unit: s

Example:
```
REFS:TIME?
```
would return, e.g.:
```
0.00125
```

Usage: Query only

Manual operation: See "Automatic adjustment" on page 67

[SENSe:]SWAPiq <State>

This command inverts the I and Q branches of the signal.

Parameters:
<State> ON | OFF | 1 | 0

Example: //Inverts the I and Q channels
```
SWAP ON
```

Manual operation: See "Inverting the I/Q branches" on page 68

[SENSe:]SWEep:LENGth <Samples>

This command defines the capture length.

Prerequisites for this command

- Turn off automatic selection of the capture time ([SENSe:]SWEep:TIME:AUTO).

Effects of this command

- Changing the capture length automatically adjusts the capture time.

Parameters:
<Samples> <numeric value>: (integer only)

Default unit: Samples

Example: //Define a capture length
```
SWE:TIME:AUTO OFF
SWE:LENG 1000000
```

Manual operation: See "Manual definition" on page 67

[SENSe:]SWEep:TIME <Time>

This command defines the capture time.

Prerequisites for this command

- Turn off automatic selection of the capture time ([SENSe:]SWEep:TIME:AUTO).

Effects of this command

- Changing the capture time automatically adjusts the capture length.

Parameters:
<Time> <numeric value>

Default unit: s

Example: //Defines a sweep time
```
SWE:TIME:AUTO OFF
SWE:TIME 10MS
```

Manual operation: See "Manual definition" on page 67

[SENSe:]SWEep:TIME:AUTO <State>

This command turns automatic selection of an appropriate capture time on and off.

When you turn on this feature, the application calculates an appropriate capture time based on the reference signal and adjusts the other acquisition settings accordingly.

Parameters:

<State> ON | OFF | 1 | 0

*RST: 1

Example: //Select automatic adjustment of the capture time
`SWE:TIME:AUTO ON`

Manual operation: See "Automatic adjustment" on page 67

TRACe:IQ:BWIDth <Bandwidth>

This command defines the analysis bandwidth with which the amplified signal is captured.

This command is available when `TRACe:IQ:SRATe:AUTO` has been turned off.

Note that when you change the analysis bandwidth, the sample rate and capture length are adjusted automatically to the new bandwidth.

Parameters:

<Bandwidth> <numeric value>

Note that the application automatically adjusts the sample rate when you change the bandwidth manually.

Default unit: Hz

Example:
`TRAC:IQ:SRAT:AUTO OFF`
`TRAC:IQ:BWID 50MHZ`
Defines a bandwidth of 50 MHz. The sample rate is adjusted accordingly.

Manual operation: See "Manual definition" on page 66

TRACe:IQ:SRATe <SampleRate>

This command defines the sample rate with which the amplified signal is captured.

This command is available when `TRACe:IQ:SRATe:AUTO` has been turned off.

Note that when you change the sample rate, the analysis bandwidth and capture length are adjusted automatically to the new sample rate.

Parameters:

<SampleRate> <numeric value>

Note that the application automatically adjusts the analysis bandwidth when you change the sample rate manually.

Default unit: Hz

Example: `TRAC:IQ:SRAT:AUTO OFF`
`TRAC:IQ:SRAT 20MHZ`
Defines a sample rate of 20 MHz. The analysis bandwidth is adjusted accordingly.

Manual operation: See "Manual definition" on page 66

TRACe:IQ:SRATe:AUTO <State>

This command turns automatic selection of an appropriate (capture) sample rate on and off.

When you turn on this feature, the application calculates an appropriate sample rate based on the reference signal and adjusts the other data acquisition settings accordingly.

Parameters:
<State> ON | OFF | 1 | 0
*RST: 1

Example: `TRAC:IQ:SRAT:AUTO ON`
Selects automatic adjustment of the sample rate.

Manual operation: See "Automatic adjustment" on page 66

TRACe:IQ:WBANd:MBWidth <Bandwidth>

This command selects the largest possible bandwidth that can be applied for the wideband signal path.

The wideband signal path is available with the corresponding bandwidth extensions available for the R&S FPS.

The command is available when you turn on `TRACe:IQ:WBANd[:STATe]`.

Example: //Restrict the bandwidth to 40 MHz
`TRAC:IQ:WBAN ON`
`TRAC:IQ:WBAN:MBW 40MHZ`

Manual operation: See "Maximum bandwidth" on page 66

TRACe:IQ:WBANd[:STATe] <State>

This command turns the wideband signal path on and off.

The wideband signal path is available with the corresponding bandwidth extensions available for the R&S FPS.

Parameters:

<State>

ON | 1
Turns on the wideband signal path.
By default, the application allows you to use the maximum available bandwidth ("Auto" mode in manual operation).
You have to turn on the wideband signal path when you want to use bandwidths greater than 40 MHz.

OFF | 0
Turns off the wideband signal path. The largest available bandwidth is 40 MHz.

Example: //Turn off the wideband signal path
`TRAC:IQ:WBAN OFF`

Manual operation: See "Maximum bandwidth" on page 66

6.6.7 Sweep Configuration

[SENSe:]SWEep:IQAVg:COUNt <Count>

Only available for backward compatibility.

Switches statistics state to "ON", sets trace mode to "IQ/Averaging" and counts to specified value.

Parameters:

<Count> <numeric value> (integer only)

Range: 1 to 10000
*RST: 1

Example: //Average over 10 data captures
`SWE:IQAV:COUN 10`

Manual operation: See "Averaging the I/Q data" on page 68

[SENSe:]SWEep:IQAVg:COUNt:CURRent?

Only available for backward compatibility.

Queries the current measurement out of a sequence of measurements that averages I/Q data.

Return values:

<Measurement> numeric value

Example: //Define number of measurements
`SWE:IQAV:COUN 10`
//Query process of measurement
`SWE:IQAV:COUN:CURR?`
would return, e.g.
`7 (out of 10)`

Usage: Query only

[SENSe:]SWEep:IQAVg:MAVerage[:STATe] <State>

Only available for backward compatibility.

Switches statistics state to "ON", sets trace mode to "IQ/Averaging" and switches continuous statistics "ON" or "OFF".

Parameters:

<State> ON | OFF | 1 | 0
*RST: 0

Example: `SWE:IQAV:MAV ON`

Manual operation: See "Averaging the I/Q data" on page 68

[SENSe:]SWEep:IQAVg[:STATe] <State>

Only available for backward compatibility.

Switches statistics state to "ON" and sets trace mode to "IQ/Averaging".

Parameters:

<State> ON | OFF | 1 | 0
*RST: OFF

Manual operation: See "Averaging the I/Q data" on page 68

[SENSe:]SWEep:STATistics[:STATe] <State>

Sets and queries the sweep statistics setting.

Parameters:

<State> ON | OFF

Example: `[SENSe]:SWEep:STATistics:STATe ON`

Manual operation: See "Statistics Table" on page 30

CONFigure:RESult:RANGe[:SELected] <ResultRange>

Sets and querys the selected result range.

Parameters:
<ResultRange> <numeric value>

Example: `CONFigure:RESult:RANGe`

Manual operation: See "Statistics Table" on page 30

6.6.8 Synchronizing Measurement Data

CONFigure:ESTimation:FULL <State>

This command turns estimation over the complete reference signal on and off.

Parameters:
<State> ON | OFF | 1 | 0

When you turn estimation over the full reference signal off, you can define an estimation range with:
•`CONFigure:ESTimation:STARt`
•`CONFigure:ESTimation:STOP`

*RST: 1

Example: //Define a synchronization range over the first 20 µs of the capture buffer
```
CONF:EST:FULL OFF
CONF:EST:STAR 0s
CONF:EST:STOP 20us
```

Manual operation: See "Defining the estimation range" on page 71

CONFigure:ESTimation:RANGe <Start>, <Stop>

This command defines start and stop values of the estimation range.

Alternatively, you can do that with

- `CONFigure:ESTimation:STARt` on page 247
- `CONFigure:ESTimation:STOP` on page 247

Setting parameters:

<Start> <numeric value>

Start time of the estimation range (relative to the beginning of the reference signal).

Default unit: s

<Stop> <numeric value>

Stop time of the estimation range (relative to the beginning of the reference signal).

Default unit: s

Example: //Define an estimation range over the first 20 µs of the reference signal

```
CONF:EST:FULL OFF
CONF:EST:RANG 0,20e-6
```

Usage: Setting only

CONFigure:ESTimation:STARt <Start>

This command defines the start value of the estimation range.

Parameters:

<Start> <numeric value>

Default unit: s

Example: See `CONFigure:ESTimation:FULL`.

Manual operation: See "Defining the estimation range" on page 71

CONFigure:ESTimation:STOP <Stop>

This command defines the end value of the estimation range.

Parameters:

<Stop> <numeric value>

Default unit: s

Example: See `CONFigure:ESTimation:FULL`.

Manual operation: See "Defining the estimation range" on page 71

CONFigure:SYNC:CONFidence <Confidence>

This command defines the synchronization confidence level.

Parameters:

<Confidence> <numeric value>

Range: 0 to 100
Default unit: PCT

Example: //Define confidence level
`CONF:SYNC:CONF 99`

Manual operation: See "Defining a synchronization confidence level" on page 71

CONFigure:SYNC:DOMain <Domain>

This command selects the synchronization method.

Parameters:

<Domain>

IQDirect
I/Q data for the reference signal is directly correlated with the reference and measured signal.

IQPDiff
Correlation on the phase differentiated I/Q data.

MAGNitude
Correlation on the magnitude of the I/Q data with no regard for phase information.

TRIGger
It is assumed that the capture is triggered at the start of the reference waveform.

*RST: IQPDiff

Example: //Try to find a correlation in the raw I/Q data
`CONF:SYNC:DOM IQD`

Manual operation: See "Selecting the synchronization method" on page 71

CONFigure:SYNC:SOFail <State>

This command turns a measurement stop on and off, when synchronization of measured and reference signal fails.

This mostly has an effect on continuous measurements. Single measurements are not affected.

Parameters:

<State> ON | OFF | 1 | 0

*RST: 0

Example: //Stop the measurement when synchronization fails
`CONF:SYNC:SOF ON`

Manual operation: See "Turning synchronization of reference and measured signal on and off" on page 70

CONFigure:SYNC:STAT <State>

This command turns synchronization between reference and measured signal on and off.

Parameters:

<State> ON | OFF | 1 | 0

*RST: 1

Example: //Turn on synchronization between reference and measured signal
`CONF:SYNC:STAT ON`

Manual operation: See "Turning synchronization of reference and measured signal on and off" on page 70

FETCh:SYNC:FAIL?

This command queries the synchronization status.

Return values:

<State> **1**
Synchronization was not successful.

0
Synchronization was successful.

Example: `FETC:SYNC:FAIL?`
would return, e.g.
`0`

Usage: Query only

6.6.9 Defining the Evaluation Range

CONFigure:EVALuation:FULL <State>

This command turns result evaluation over the complete capture buffer on and off.

Parameters:

<State> ON | OFF | 1 | 0

When you turn calculation over the full capture buffer off, you can define an evaluation range with:

- `CONFigure:EVALuation:STARt`
- `CONFigure:EVALuation:STOP`

*RST: 1

Example: //Define an evaluation range over 45 µs of the capture buffer

```
CONF:EVAL:FULL OFF
CONF:EVAL:STAR 5us
CONF:EVAL:STOP 50us
```

Manual operation: See "Defining the evaluation range" on page 73

CONFigure:EVALuation:RANGe <Start>, <Stop>

This command defines start and stop values of the evaluation range.

Alternatively, you can do that with

- `CONFigure:EVALuation:STARt` on page 250
- `CONFigure:EVALuation:STOP` on page 250

Setting parameters:

<Start> <numeric value>

Start time of the evaluation range (relative to the beginning of the reference signal).

Default unit: s

<Stop> <numeric value>

Stop time of the evaluation range (relative to the beginning of the reference signal).

Default unit: s

Example: //Define an evaluation range over 45 µs of the reference signal, beginning at 5 µs into the signal

```
CONF:EVAL:FULL OFF
CONF:EVAL:RANG 5e-6,50e-6
```

Usage: Setting only

CONFigure:EVALuation:STARt <Start>

This command defines the start value of the evaluation range.

Parameters:

<Start> <numeric value>

Default unit: s

Example: See `CONFigure:EVALuation:FULL`.

Manual operation: See "Defining the evaluation range" on page 73

CONFigure:EVALuation:STOP <Stop>

This command defines the end value of the evaluation range.

Parameters:

<Stop> <numeric value>

Default unit: s

Example: See `CONFigure:EVALuation:FULL`.

Manual operation: See "Defining the evaluation range" on page 73

6.6.10 Estimating and Compensating Signal Errors

6.6.10.1 Error Estimation and Compensation

CONFigure:SIGNal:ERRor:COMPensation:ADRoop[:STATe] <State>

This command turns compensation of the amplitude droop on and off.

Prerequisites for this command

- Turn on estimation of sample rate (`CONFigure:SIGNal:ERRor:ESTimation:ADRoop[:STATe]`).

Parameters:

<State>   ON | OFF | 1 | 0
    *RST: 1

Example: //Turn on error compensation
`CONF:SIGN:ERR:COMP:ADR ON`

CONFigure:SIGNal:ERRor:COMPensation:FERRor[:STATe] <State>

This command turns compensation of the frequency error on and off.

Prerequisites for this command

- Turn on estimation of sample rate (`CONFigure:SIGNal:ERRor:ESTimation:FERRor[:STATe]`).

Parameters:

<State>   ON | OFF | 1 | 0
    *RST: 1

Example: //Turn on error compensation
`CONF:SIGN:ERR:COMP:FERR ON`

CONFigure:SIGNal:ERRor:COMPensation:IQIMbalance[:STATe] <State>

This command turns compensation of the I/Q imbalance on and off.

Prerequisites for this command

- Turn on estimation of sample rate (`CONFigure:SIGNal:ERRor:ESTimation:IQIMbalance[:STATe]`).

Parameters:

<State> ON | OFF | 1 | 0

*RST: 1

Example: //Turn on error compensation
`CONF:SIGN:ERR:COMP:IQIM ON`

CONFigure:SIGNal:ERRor:COMPensation:IQOFfset[:STATe] <State>

This command turns compensation of the I/Q offset on and off.

Prerequisites for this command

- Turn on estimation of sample rate (`CONFigure:SIGNal:ERRor:ESTimation:IQOFfset[:STATe]`).

Parameters:

<State> ON | OFF | 1 | 0

*RST: 1

Example: //Turn on error compensation
`CONF:SIGN:ERR:COMP:IQOF ON`

CONFigure:SIGNal:ERRor:COMPensation:SRATe[:STATe] <State>

This command turns compensation of the sample rate error on and off.

Prerequisites for this command

- Turn on estimation of sample rate (`CONFigure:SIGNal:ERRor:ESTimation:SRATe[:STATe]`).

Parameters:

<State> ON | OFF | 1 | 0

*RST: 1

Example: //Turn on error compensation
`CONF:SIGN:ERR:COMP:SRAT ON`

CONFigure:SIGNal:ERRor:ESTimation:ADRoop[:STATe] <State>

This command turns estimation of the amplitude droop on and off.

Parameters:

<State> ON | OFF | 1 | 0

*RST: 1

Example: //Turn on error estimation
`CONF:SIGN:ERR:EST:ADR ON`

CONFigure:SIGNal:ERRor:ESTimation:FERRor[:STATe] <State>

This command turns estimation of the frequency error on and off.

Parameters:

<State> ON | OFF | 1 | 0

*RST: 1

Example: //Turn on error estimation
`CONF:SIGN:ERR:EST:FERR ON`

CONFigure:SIGNal:ERRor:ESTimation:IQIMbalance[:STATe] <State>

This command turns estimation of the I/Q imbalance on and off.

Parameters:

<State> ON | OFF | 1 | 0

*RST: 1

Example: //Turn on error estimation
`CONF:SIGN:ERR:EST:IQIM ON`

CONFigure:SIGNal:ERRor:ESTimation:IQOFfset[:STATe] <State>

This command turns estimation of the I/Q offset on and off.

Parameters:

<State> ON | OFF | 1 | 0

*RST: 1

Example: //Turn on error estimation.
`CONF:SIGN:ERR:EST:IQOF ON`

CONFigure:SIGNal:ERRor:ESTimation:SRATe[:STATe] <State>

This command turns estimation of the sample rate error on and off.

Parameters:

<State> ON | OFF | 1 | 0

*RST: 1

Example: //Turn on error estimation
`CONF:SIGN:ERR:EST:SRAT ON`

6.6.10.2 Equalizer

CONFigure:EQUalizer:FILTer:FILE:FORMat <Source>

This command selects the file format to which the equalizer filter is exported.

Parameters:

<Source> **CSV**
Filter is written to a `csv` file.

FRES
Filter is written to a `fres` file.

*RST: CSV

Example: //Select file format for equalizer filter
`CONF:EQU:FILT:FILE:FORM CSV`

Manual operation: See "Using the equalizer" on page 75

CONFigure:EQUalizer:FILTer:LENGth <Length>

This command defines the length of the filter that the equalizer training is based on.

Parameters:

<Length> <numeric value> (integer only)

Example: //Define equalizer filter length
`CONF:EQU:FILT:LENG 25`

Manual operation: See "Using the equalizer" on page 75

CONFigure:EQUalizer:FPARameters <Coefficient>...

This command defines the filter coefficients.

You can use this command to define the filter coefficients manually instead of training a filter.

Parameters:

<Coefficient> <numeric value> (integer only)

List of comma separated values.
Each coefficient consists of a real and an imaginary value.
`<Coefficient_1_I>,<Coefficient_2_Q>,`
`<Coefficient_2_I>,<Coefficient_2_Q>,...,`
`<Coefficient_n_I>,<Coefficient_n_Q>`

Example: //Define a filter with a length of five, number of values therefore must be 10
`CONF:EQU:FPAR 5,8,5,10,10,12,5,2,2,1`

Manual operation: See "Using the equalizer" on page 75

CONFigure:EQUalizer[:STATe] <State>

This command turns the equalizer on and off.

Prerequisites for this command

- Load equalizer filter data (either by training or by restoring a file with equalizer information).
  – `CONFigure:EQUalizer:TRAin`
  – `MMEMory:LOAD:EQUalizer:FILTer:COEFficient`

Parameters:

<State> ON | OFF | 1 | 0
*RST: OFF

Example: //Turn on equalizer
`MMEM:LOAD:EQU:FILT:COEF 'c:\filter.csv'`
`CONF:EQU ON`

Manual operation: See "Using the equalizer" on page 75

CONFigure:EQUalizer:TRAin

This command initiates a training sequence for the equalizer filter.

Note that you have to synchronize the measurement before you can initiate a training sequence.

Prerequisites for this command

- Define a filter length (`CONFigure:EQUalizer:FILTer:LENGth`).

Usage: Event

Manual operation: See "Using the equalizer" on page 75

MMEMory:LOAD:EQUalizer:FILTer:COEFficient <FileName>

This command restores an equalizer filter that you have previously saved.

Setting parameters:

<FileName> String containing the file name and location of the filter (csv file format).

Example: //Restore filter file
`MMEM:LOAD:EQU:FILT:COEF 'C:\filter.csv'`

Usage: Setting only

Manual operation: See "Using the equalizer" on page 75

MMEMory:STORe:EQUalizer:FILTer:COEFficient <FileName>

This command stores the equalizer filter that has been calculated.

Prerequisites for this command

- Train an equalizer filter (`CONFigure:EQUalizer:TRAin`).

Setting parameters:

<FileName> String containing the file name and location of the filter (csv file format).

Example: //Store filter file
```
CONF:EQU:TRA
MMEM:STOR:EQU:FILT:COEF 'C:\filter.csv'
```

Usage: Setting only

Manual operation: See "Using the equalizer" on page 75

6.6.11 Applying a System Model

CONFigure:MODeling:AMAM:ORDer <Order>

This command defines the order (or degree) of the AM/AM model polynomials that are calculated by the application.

Parameters:

<Order> String containing the polynomials to be calculated.
You can either select a range of polynomials (e.g. "1-7"), a selection of polynomials (e.g. "1,3,5") or a combination of both (e.g. "1,3-5").

Range: 0 to 18
*RST: "0-7"

Example: //Calculate the polynomials to the 1st, 2nd, 3rd, 4th and 5th degree
```
CONF:MOD:AMAM:ORD "1-5"
```

Example: //Calculate the polynomials to the 1st, 3rd and 5th degree
```
CONF:MOD:AMAM:ORD "1,3,5"
```

Manual operation: See "Selecting the degree of the polynomial" on page 78

CONFigure:MODeling:AMPM:ORDer <Order>

This command defines the order (or degree) of the AM/PM model polynomials that are calculated by the application.

Parameters:

<Order> String containing the polynomials to be calculated.
You can either select a range of polynomials (e.g. "1-7"), a selection of polynomials (e.g. "1,3,5") or a combination of both (e.g. "1,3-5").

Range: 0 to 18
*RST: "1-7"

Example: //Calculate the polynomials to the 1st, 3rd, 4th and 5th degree
`CONF:MOD:AMPM:ORD "1,3-5"`

Manual operation: See "Selecting the degree of the polynomial" on page 78

CONFigure:MODeling:LRANge <Level>

This command defines the modeling level range.

Parameters:

<Level> <numeric value>

Default unit: dB

Example: //Define a modeling level range
`CONF:MOD:LRAN 30`

Manual operation: See "Defining the modeling range" on page 78

CONFigure:MODeling:NPOints <Points>

This command defines the number of modeling points.

Parameters:

<Points> <numeric value>: (integer only)

*RST: 50
Default unit: ---

Example: //Calculate the model based on 50 points
`CONF:MOD:NPO 50`

Manual operation: See "Defining the modeling range" on page 78

CONFigure:MODeling:SCALe <State>

This command selects the method by which the input power range is split into smaller ranges for the calculation of the amplifier model.

Parameters:

<State> **LINear**
Input power range is split on a linear basis.

LOGarithmic
Input power range is split on a logarithmic basis.
*RST: LOGarithmic

Example: //Apply a linear scale for the model calculation
`CONF:MOD:SCAL LIN`

Manual operation: See "Selecting the modeling scale" on page 79

CONFigure:MODeling:SEQuence <State>

This command selects the sequence in which the models are calculated.

Parameters:
<State>

AMFirst
Calculates the AM/AM model before calculating the AM/PM model.

PMFirst
Calculates the AM/PM model before calculating the AM/AM model.

*RST: AMFirst

Example: //Calculate AM/AM model first
`CONF:MOD:SEQ AMF`

Manual operation: See "Turning system modeling on and off" on page 77

CONFigure:MODeling[:STATe] <State>

This command turns system modeling on and off.

Parameters:
<State> ON | OFF | 1 | 0
*RST: 0

Example: //Turn on system modeling
`CONF:MOD ON`

Manual operation: See "Turning system modeling on and off" on page 77

6.6.12 Applying Digital Predistortion

CONFigure:DDPD:ABORt

This command stops a DPD sequence and discards the predistorted I/Q data that have been calculated.

Prerequisites for this command

- Turn on direct DPD (`CONFigure:DDPD[:STATe]` on page 262).
- Initiate a DPD sequence (`CONFigure:DDPD:STARt` on page 261).

Example: //Stop a DPD sequence
`CONF:DDPD:ABOR`

Usage: Event

CONFigure:DDPD:APPLy[:STATe] <State>

This command transfers the waveform file with the correction values to the signal generator and applies them to the input signal.

Prerequisites for this command

- Turn on direct DPD (`CONFigure:DDPD[:STATe]`).
- Run a DPD sequence (`CONFigure:DDPD:STARt`).

Parameters:
<State> ON | OFF | 1 | 0
*RST: OFF

Example: //Run a DPD sequence and transfer the correction value to the generator
`CONF:DDPD ON`
`CONF:DDPD:STAR`
`CONF:DDPD:APP ON`

Manual operation: See "Running a direct DPD sequence" on page 84

CONFigure:DDPD:COUNt <Count>

This command defines the number of iterations in a direct DPD sequence.

Prerequisites for this command

- Turn on direct DPD (`CONFigure:DDPD[:STATe]`).

Parameters:

<Count> — <numeric value> (integer only)

Range: 1 to 1000
*RST: 10

Example: //Define number of iterations
`CONF:DDPD:COUN 25`

Manual operation: See "Running a direct DPD sequence" on page 84

CONFigure:DDPD:COUNt:CURRent?

This command queries the process of the direct DPD sequence (number of current iteration).

- Turn on direct DPD (`CONFigure:DDPD[:STATe]`).
- Start a DPD sequence (`CONFigure:DDPD:STARt`).

Return values:

<Iterations>

Example: //Define number of iterations
`CONF:DDPD:COUN 10`
//Query process of measurement
`CONF:DDPD:COUN:CURR?`
would return, e.g.
`7 (out of 10)`

Usage: Query only

CONFigure:DDPD:FINish

This command stops a DPD sequence before all iterations are done and keeps the predistorted I/Q data that have been calculated.

Prerequisites for this command

- Turn on direct DPD (`CONFigure:DDPD[:STATe]`).

- Initiate a DPD sequence (CONFigure:DDPD:STARt).

Example: //Stop a DPD sequence
CONF:DDPD:FIN

Usage: Event

CONFigure:DDPD:GEXPansion <GainExpansion>

This command sets the gain expansion for Direct DPD.

Parameters:
<GainExpansion> <numeric value>
Default unit: dB

Example: //Define gain expansion
CONFigure:DDPD:GEXPansion 2

Manual operation: See "Running a direct DPD sequence" on page 84

CONFigure:DDPD:FNAMe <FileName>

This command defines a file name for the I/Q file that contains the predistorted I/Q data that was generated by the direct DPD.

Prerequisites for this command

- Turn on direct DPD (CONFigure:DDPD[:STATe]).

Parameters:
<FileName> String containing the file name (including file type .wv).

Example: //Define file name of direct DPD file
CONF:DDPD:FNAM 'DirectDPD.wv'

Manual operation: See "Running a direct DPD sequence" on page 84

CONFigure:DDPD:STARt

This command initiates a direct DPD sequence with the number of iterations you have defined.

You can define the number of iterations with CONFigure:DDPD:COUNt.

Prerequisites for this command

- Turn on direct DPD (CONFigure:DDPD[:STATe] on page 262).

Example: //Initiate direct DPD sequence
CONF:DDPD:STAR

Usage: Event

Manual operation: See "Running a direct DPD sequence" on page 84

CONFigure:DDPD[:STATe] <State>

This command selects the type of DPD.

Parameters:

<State>

ON | 1
Selects direct DPD.

OFF | 0
Selects polynomial DPD.

*RST: OFF

Example: //Select direct DPD
`CONF:DDPD ON`

CONFigure:DDPD:TRADeoff <Power Linearity Tradeoff>

This command defines the power / linearity tradeoff for direct DPD calculation.

Prerequisites for this command

- Turn on direct DPD (`CONFigure:DDPD[:STATe]`).

Parameters:

<Power Linearity Tradeoff> <numeric value>

Default unit: PCT

Example: //Define linearity tradeoff
`CONF:DDPD:TRAD 75`

Manual operation: See "DPD Power / Linearity Tradeoff" on page 82

CONFigure:DPD:AMAM:LEDState?

This command queries the state of the calculation of the AM/AM distortion curve.

Return values:

<State>

GREen
Calculation was successful.

GREY
Unknown calculation state.

RED
Calculation was not successful.

Example: `CONF:DPD:AMAM:LEDS?`
would return, e.g.:
`RED`

Usage: Query only

CONFigure:DPD:AMAM[:STATe] <State>

This command turns AM/AM predistortion on and off.

Prerequisites for this command

- Turn on polynomial DPD (CONFigure:DDPD[:STATe]).

Parameters:

<State>	ON | OFF | 1 | 0 *RST: 1

Example: //Calculate AM/AM curve
`CONF:DPD:AMAM ON`

Manual operation: See "Selecting the order of model calculation" on page 81

CONFigure:DPD:AMPM:LEDState?

This command queries the state of the calculation of the AM/PM distortion curve.

Return values:

<State>

GREen
Calculation was successful.

GREY
Unknown calculation state.

RED
Calculation was not successful.

Example: `CONF:DPD:AMPM:LEDS?`
would return, e.g.:
`RED`

Usage: Query only

CONFigure:DPD:AMPM[:STATe] <State>

This command turns AM/PM predistortion on and off.

Prerequisites for this command

- Turn on polynomial DPD (CONFigure:DDPD[:STATe]).

Parameters:

<State>	ON | OFF | 1 | 0 *RST: 1

Example: //Calculate AM/PM curve
`CONF:DPD:AMPM ON`

Manual operation: See "Selecting the order of model calculation" on page 81

CONFigure:DPD:AMXM[:STATe] <State>

This command turns AM/AM and AM/PM predistortion on and off (at the same time).

Alternatively, you can do that with:

- CONFigure:DPD:AMAM[:STATe]

and

- CONFigure:DPD:AMPM[:STATe]

However, using CONFigure:DPD:AMXM[:STATe] is the smoother way.

Prerequisites for this command

- Turn on polynomial DPD (CONFigure:DDPD[:STATe]).

Setting parameters:

<State> ON | OFF | 1 | 0

Example: //Calculate both AM/AM and AM/PM predistortion
CONF:DPD:AMXM ON

Usage: Setting only

Manual operation: See "Selecting the order of model calculation" on page 81

CONFigure:DPD:FILE:GENerate

This command generates the waveform files containing predistortion information within the amplifier application.

All in all, the command generates three waveform files: AM/AM only, AM/PM only and AM/AM plus AM/PM.

It also transfers these waveform files to the connected signal generator.

Prerequisites for this command

- Turn on polynomial DPD (CONFigure:DDPD[:STATe]).

Example: //Calculate DPD within the amplifier application and transfer the result to the signal generator
CONF:DPD:METH WFIL
CONF:DPD:FILE:GEN

Usage: Event

CONFigure:DPD:FILE:GENerate:ALL

This command generates the waveform files containing predistortion information within the amplifier application.

All in all, the command generates three waveform files: AM/AM only, AM/PM only and AM/AM plus AM/PM.

It also transfers these waveform files to the connected signal generator and turns on the AM/AM and AM/PM DPDs.

Alternatively, you can do that with:

- CONFigure:DPD:FILE:GENerate
  and
- CONFigure:DPD:AMXM[:STATe] on page 263

Prerequisites for this command

- Turn on polynomial DPD (CONFigure:DDPD[:STATe]).

Example: //Calculate DPD within the amplifier application and transfer the result to the signal generator
`CONF:DPD:METH WFIL`
`CONF:DPD:FILE:GEN`

Usage: Event

CONFigure:DPD:FNAMe <FileName>

This command defines a name for the DPD correction table.

Prerequisites for this command

- Turn on polynomial DPD (CONFigure:DDPD[:STATe]).

Parameters:

<FileName> String containing the DPD table file name.

Example: //Defines the DPD table name
`CONF:DPD:FNAM 'DPDTable'`

Manual operation: See "Selecting the DPD shaping method" on page 81

CONFigure:DPD:METHod <Method>

This command selects the method with which the application determines the DPD.

Prerequisites for this command

- Turn on polynomial DPD (CONFigure:DDPD[:STATe]).

Parameters:

<Method>

GENerator
Signal generator applies the DPD parameters calculated by the amplifier application to the generated RF signal in real-time.
Option R&S SMW-K541 is required on the generator for this method.

WFILe
Signal generator applies the DPD to the generated RF signal through a waveform file.
No additional equipment is required on the signal generator for this method.
Use CONFigure:DPD:FILE:GENerate to actually generate the DPD and transfer it to the generator.

*RST: GENerator

Example: //Calculates the DPD within the amplifier application
`CONF:DPD:METH WFIL`

Manual operation: See "Selecting the DPD method" on page 80

CONFigure:DPD:SEQuence <State>

This command selects the order in which the AM/AM and AM/PM distortion are applied.

Available when both have been turned on.

Prerequisites for this command

- Turn on polynomial DPD (`CONFigure:DDPD[:STATe]`).
- Turn on both AM/AM and AM/PM calculation (`CONFigure:DPD:AMAM[:STATe]` / `CONFigure:DPD:AMPM[:STATe]`).

Parameters:

<State>

AMFirst
Calculates the AM/AM distortion first, then the AM/PM distortion.

PMFirst
Calculates the AM/PM distortion first, then the AM/AM distortion.

Example:
//Calculates the AM/AM curve first
`CONF:DPD:SEQ AMF`

Manual operation: See "Selecting the order of model calculation" on page 81

CONFigure:DPD:SHAPing:MODE <Method>

This command selects the method use to shape the DPD function.

Parameters:

<Method>

POLYnomial
DPD function based on the characteristics of the polynomial system model.

TABLe
DPD function based on the correction values kept in a table calculated by the R&S SMW.

*RST: TABLe

Example:
//Select DPD shaping method
`CONF:DPD:SHAP:MODE TABL`

Manual operation: See "Selecting the DPD shaping method" on page 81

CONFigure:DPD:TRADeoff <Power Linearity Tradeoff>

This command defines the power / linearity tradeoff for polynomial DPD calculation.

Prerequisites for this command

- Turn on polynomial DPD (`CONFigure:DDPD[:STATe]`).

Parameters:

<Power Linearity Tradeoff>

<numeric value>

Default unit: PCT

Example: //Define linearity tradeoff
`CONF:DPD:TRAD 75`

Manual operation: See "DPD Power / Linearity Tradeoff" on page 82

CONFigure:DPD:UPDate

This command updates the DPD shaping tables on the R&S SMW when new measurement data is available.

Prerequisites for this command

- Turn on polynomial DPD (`CONFigure:DDPD[:STATe]`).

Example: //Update the shaping table
`CONF:DPD:UPD`

Usage: Event

CONFigure:DPD:UPDate:ALL

This command updates the DPD shaping tables on the R&S SMW when new measurement data is available.

In addition, this command also turns on the DPD (AM/AM **and** AM/PM).

Using one command only to do those things has the advantage of a slightly shorter execution time.

Alternatively, you can do that with:

- `CONFigure:DPD:UPDate` on page 267
  and
- `CONFigure:DPD:AMXM[:STATe]` on page 263

Prerequisites for this command

- Turn on polynomial DPD (`CONFigure:DDPD[:STATe]`).

Example: //Update the tables and turn on AM/AM and AM/PM predistortion
`CONF:DPD:UPD:ALL`

Usage: Event

CONFigure:DPD:UPDate:LEDState?

This command queries the state of the DPD calculation.

The information of the return result depends on the DPD method:

- DPD calculated by the generator (with option K541): Query of the state of the update of the shaping table or the polynomial coefficients.
- DPD calculation by the Amplifier application: Query of the state of waveform file generation and its upload to the generator.

Prerequisites for this command

- Turn on polynomial DPD (CONFigure:DDPD[:STATe]).

Return values:

<State>

GREen
Transmission was successful.

GREY
Unknown transmission state.

RED
Transmission was not successful.

Example:

//Query LED state
`CONF:DPD:UPD`
`CONF:DPD:UPD:LEDS?`
would return, e.g.:
`GREY`

Usage: Query only

FETCh:DDPD:OPERation:STATus?

This command queries the state of a direct DPD operation.

Return values:

<State> ON | OFF | 1 | 0

ON
Direct DPD operation was successful.

OFF
Direct DPD operation was not successful.

Example:

//Query direct DPD state
`FETC:DDPD:OPER:STAT?`

Usage: Query only

FETCh:DPD:POLYnomial?

This command queries the polynomial factors of the correctional polynomial.

Prerequisites for this command

- Turn on polynomial DPD (CONFigure:DDPD[:STATe]).
- Run polynomial DPD (CONFigure:DPD:FILE:GENerate).

Return values:

<Values>

List of numerical values.
The number of values depends on the DPD configuration.
The real and imaginary parts of the DPD coefficients are returned interleaved in the following order: real(a0), imag(a0), real(a1), imag(a1), …

Example:

//Query polynomial factors
`FETC:DPD:POLY?`

Usage: Query only

MMEMory:STORe:DDPD <FileName>

This command stores the direct DPD information in a file.

Prerequisites for this command

- Turn on direct DPD (`CONFigure:DDPD[:STATe]` on page 262).
- Run a DPD sequence (`CONFigure:DDPD:STARt` on page 261).

Setting parameters:

<FileName> String containing the file name and location of the file.

Example: //Run a DPD sequence and save the DPD

```
CONF:DDPD ON
CONF:DDPD:STAR
MMEM:STOR:DDPD 'c:\directdpd.wv'
```

Usage: Setting only

Manual operation: See "Running a direct DPD sequence" on page 84

MMEMory:STORe:DPD <FileName>

This command generates and stores a waveform containing the DPD in a file you have specified.

Prerequisites for this command

- DPD method "Generate Predistorted Waveform File" has to be selected (`CONFigure:DPD:METHod` = WFILe)
- The DPD calculation has been initiated with `CONFigure:DPD:FILE:GENerate`.

Setting parameters:

<FileName> String containing the file name.

Example:

```
CONF:DPD:METH WFIL
CONF:DPD:FILE:GEN
MMEM:STOR:DPD 'DPD_WV'
```

Calculates the DPD within the Amplifier application, transfers the result to the signal generator and saves it in a file.

Usage: Setting only

6.6.13 Configuring ACLR Measurements

CALCulate<n>:MARKer<m>:FUNCtion:POWer:RESult? <Item>

This command queries the (numerical) results of the ACLR measurement.

Suffix:

<n>	Window
<m>	irrelevant

Query parameters:

<Item>

ACP
Queries the results of the ACLR measurement.
Returns the power for every active transmission and adjacent channel. The order is:
• power of the transmission channels
• power of adjacent channel (lower,upper)

Example:

```
CALC:MARK:FUNC:POW:RES?
```
would return, e.g.
```
-21.76, 3.21, 2.57
```

Usage: Query only

Manual operation: See "Adjacent Channel Leakage Error (ACLR)" on page 11

[SENSe:]POWer:ACHannel:ACPairs <ChannelPairs>

This command defines the number of pairs of adjacent and alternate channels.

Parameters:

<ChannelPairs>

Range: 0 to 12
*RST: 1

Manual operation: See " Number of channels: Tx , Adj " on page 87

[SENSe:]POWer:ACHannel:AABW <State>

This command turns automatic selection of the measurement bandwidth for ACLR measurements on and off.

When you turn this on, the application selects a measurement bandwidth that is large enough to capture all channels evaluated by the ACLR measurement.

Parameters:
<State> ON | OFF | 1 | 0

Example: //Turn on automatic selection of the measurement bandwidth
`POW:ACH:AABW ON`

Manual operation: See "Selecting the measurement bandwidth" on page 87

[SENSe:]POWer:ACHannel:BANDwidth:ACHannel <Bandwidth>

This command defines the channel bandwidth of the adjacent channels.

The adjacent channels are the first channels to the left and right of the transmission channels. If you set the channel bandwidth for these channels, the R&S FPS sets the bandwidth of the alternate channels to the same value (not for MSR signals).

Parameters:
<Bandwidth> Range: 100 Hz to 1000 MHz
*RST: 14 kHz
Default unit: Hz

Manual operation: See " Channel Bandwidth " on page 88

[SENSe:]POWer:ACHannel:BANDwidth:ALTernate<ch> <Bandwidth>

This command defines the channel bandwidth of the alternate channels.

Suffix:
<ch> 1..n
Alternate channel number

Parameters:
<Bandwidth> Range: 100 Hz to 1000 MHz
*RST: 14 kHz
Default unit: Hz

Manual operation: See " Channel Bandwidth " on page 88

[SENSe:]POWer:ACHannel:BANDwidth[:CHANnel<ch>] <Bandwidth>

This command defines the channel bandwidth of the transmission channels.

Suffix:
<ch> 1..n
Tx channel number

Parameters:
<Bandwidth> Range: 100 Hz to 1000 MHz
*RST: 14 kHz
Default unit: Hz

Manual operation: See " Channel Bandwidth " on page 88

[SENSe:]POWer:ACHannel:FILTer:ALPHa:ACHannel <Alpha>

This command defines the roll-off factor for the adjacent channel weighting filter.

Parameters:

<Alpha> Roll-off factor

Range: 0 to 1
*RST: 0.22

Manual operation: See " Weighting Filters " on page 89

[SENSe:]POWer:ACHannel:FILTer:ALPHa:ALTernate<ch> <Alpha>

This command defines the roll-off factor for the alternate channel weighting filter.

Suffix:

<ch> 1..n
Alternate channel number

Parameters:

<Alpha> Roll-off factor

Range: 0 to 1
*RST: 0.22

Manual operation: See " Weighting Filters " on page 89

[SENSe:]POWer:ACHannel:FILTer:ALPHa:CHANnel<ch> <Alpha>

This command defines the roll-off factor for the transmission channel weighting filter.

Suffix:

<ch> 1..n
Tx channel number

Parameters:

<Alpha> Roll-off factor

Range: 0 to 1
*RST: 0.22

Manual operation: See " Weighting Filters " on page 89

[SENSe:]POWer:ACHannel:FILTer[:STATe]:ACHannel <State>

This command turns the weighting filter for the adjacent channel on and off.

Parameters:

<State> ON | OFF | 1 | 0

*RST: 0

Manual operation: See " Weighting Filters " on page 89

[SENSe:]POWer:ACHannel:FILTer[:STATe]:ALTernate<ch> <State>

This command turns the weighting filter for an alternate channel on and off.

Suffix:
<ch> 1..n
Alternate channel number

Parameters:
<State> ON | OFF | 1 | 0
*RST: 0

Manual operation: See " Weighting Filters " on page 89

[SENSe:]POWer:ACHannel:FILTer[:STATe]:CHANnel<ch> <State>

This command turns the weighting filter for a transmission channel on and off.

Suffix:
<ch> 1..n
Tx channel number

Parameters:
<State> ON | OFF | 1 | 0
*RST: 0

Manual operation: See " Weighting Filters " on page 89

[SENSe:]POWer:ACHannel:REFerence:TXCHannel:AUTO <RefChannel>

This command selects the reference channel for relative measurements.

You need at least one channel for the command to work.

Parameters:
<RefChannel> MINimum | MAXimum | LHIGhest

MINimum
Transmission channel with the lowest power

MAXimum
Transmission channel with the highest power

LHIGhest
Lowest transmission channel for lower adjacent channels and highest transmission channel for upper adjacent channels

Example: `POW:ACH:REF:TXCH:AUTO MAX`
Selects the channel with the peak power as reference channel.

Manual operation: See " Reference Channel " on page 87

[SENSe:]POWer:ACHannel:REFerence:TXCHannel:MANual <ChannelNumber>

This command defines a reference channel for relative ACLR measurements.

You need at least one channel for the command to work.

Parameters:
<ChannelNumber> Range: 1 to 18
*RST: 1

Manual operation: See " Reference Channel " on page 87

[SENSe:]POWer:ACHannel:SPACing:CHANnel<ch> <Spacing>

This command defines the distance between transmission channels.

If you set the channel spacing for a transmission channel, the R&S FPS sets the spacing of the lower transmission channels to the same value, but not the other way round. The command works hierarchically: to set a distance between the 2nd and 3rd and 3rd and 4th channel, you have to set the spacing between the 2nd and 3rd channel first.

Suffix:
<ch> 1..n
Tx channel number

Parameters:
<Spacing> Range: 14 kHz to 2000 MHz
*RST: 20 kHz
Default unit: Hz

Manual operation: See " Channel Spacings " on page 88

[SENSe:]POWer:ACHannel:SPACing[:ACHannel] <Spacing>

This command defines the distance from transmission channel to adjacent channel.

A change of the adjacent channel spacing causes a change in the spacing of all alternate channels below the adjacent channel.

Parameters:
<Spacing> Range: 100 Hz to 2000 MHz
*RST: 14 kHz
Default unit: Hz

Manual operation: See " Channel Spacings " on page 88

[SENSe:]POWer:ACHannel:SPACing:ALTernate<ch> <Spacing>

This command defines the distance from transmission channel to alternate channels.

Suffix:
<ch> 1..n
Alternate channel number

Parameters:
<Spacing> Range: 100 Hz to 2000 MHz
*RST: 40 kHz (ALT1), 60 kHz (ALT2), 80 kHz (ALT3), ...
Default unit: Hz

Manual operation: See " Channel Spacings " on page 88

[SENSe:]POWer:ACHannel:TXCHannel:COUNt <Number>

This command defines the number of transmission channels.

The command works for measurements in the frequency domain.

Parameters:

<Number> Range: 1 to 18
*RST: 1

Manual operation: See " Number of channels: Tx , Adj " on page 87

6.6.14 Configuring Power Measurements

CONFigure:POWer:RESult:P3DB:REFerence <RefPower>

This command defines the input power corresponding to the gain reference required to calculate the compression points.

Prerequisites for this command

- Turn off automatic calculation of the reference point (`CONFigure:POWer:RESult:P3DB[:STATe]`).

Parameters:

<RefPower> <numeric value>
Default unit: dBm

Example: //Reference point is the gain measured at an input power of 3 dBm
```
CONF:POW:RES:P3DB OFF
CONF:POW:RES:P3DB:REF 3
```

Manual operation: See "Configuring compression point calculation" on page 85

CONFigure:POWer:RESult:P3DB[:STATe] <State>

This command turns automatic calculation of the reference point required to determine the compression points (1 dB, 2 dB and 3 dB) on and off.

Parameters:

<State> ON | OFF | 1 | 0
*RST: 1

Example: //Automatically determine the reference point
```
CONF:POW:RES:P3DB ON
```

Manual operation: See "Configuring compression point calculation" on page 85

6.6.15 Configuring Parameter Sweeps

CONFigure:PSWeep:ADJust:LEVel[:STATe] <State>

This command turns synchronization of the generator output level and the analyzer reference level on and off.

When you synchronize the levels, it is recommended to also define the expected gain of the DUT with `CONFigure:PSWeep:EXPected:GAIN`.

Prerequisites for this command

- Select "Generator Power" as one of the parameters.

Parameters:

<State> ON | OFF | 1 | 0

Example: //Synchronize the generator output level and the analyzer reference level
`CONF:PSW:ADJ:LEV ON`

Manual operation: See "Synchronizing the levels of signal generator and analyzer" on page 91

CONFigure:PSWeep:EXPected:GAIN <Gain>

This command defines the expected gain of the DUT.

This is necessary when you synchronize the generator output level and the reference level of the analyzer `CONFigure:PSWeep:ADJust:LEVel[:STATe]` = ON.

Prerequisites for this command

- Select "Generator Power" as one of the parameters.

Parameters:

<Gain> <numeric value>

Default unit: dB

Example: //Define expected gain
`CONF:PSW:ADJ:LEV ON`
`CONF:PSW:EXP:GAIN 5`

Manual operation: See "Synchronizing the levels of signal generator and analyzer" on page 91

CONFigure:PSWeep[:STATe] <State>

This command turns the parameter sweep on and off.

Parameters:

<State> ON | OFF | 1 | 0

Example: //Turn on parameter sweep
`CONF:PSW ON`

Manual operation: See "Turning the parameter sweep on and off" on page 90

CONFigure:PSWeep:X:SETTing <Setting>

This command selects the parameter type for the first parameter controlled by the parameter sweep.

Parameters:

<Setting>

BIAS
Controls the envelope bias.

DELay
Controls the delay between envelope and RF signal.

FREQuency
Controls the frequency.

POWer
Controls the output level.

Example: See `CONFigure:PSWeep:Y:SETTing`.

CONFigure:PSWeep:X:STARt <Start>

This command defines the start value for the first parameter controlled by the parameter sweep.

Parameters:

<Start> <numeric value> whose unit depends on the parameter type you have selected with `CONFigure:PSWeep:Y:SETTing`:
• Hz in case of the center frequency
• dBm in case of the output level
• s in case of the delay between envelope and RF signal
• V in case of the envelope bias

Example: See `CONFigure:PSWeep:Y:SETTing`.

CONFigure:PSWeep:X:STEP <Step>

This command defines the stepsize for the first parameter controlled by the parameter sweep.

Parameters:

<Step> <numeric value> whose unit depends on the parameter type you have selected with `CONFigure:PSWeep:Y:SETTing`:
- Hz in case of the center frequency
- dB in case of the output level
- s in case of the delay between envelope and RF signal
- V in case of the envelope bias

Example: See `CONFigure:PSWeep:Y:SETTing`.

CONFigure:PSWeep:X:STOP <Stop>

This command defines the stop value for the first parameter controlled by the parameter sweep.

Parameters:

<Stop> <numeric value> whose unit depends on the parameter type you have selected with `CONFigure:PSWeep:Y:SETTing`:
- Hz in case of the center frequency
- dBm in case of the output level
- s in case of the delay between envelope and RF signal
- V in case of the envelope bias

Example: See `CONFigure:PSWeep:Y:SETTing`.

CONFigure:PSWeep:Y:SETTing <Setting>

This command selects the parameter type for the second parameter controlled by the parameter sweep.

Parameters:

<Setting>

BIAS
Controls the envelope bias.

DELay
Controls the delay between envelope and RF signal.

FREQuency
Controls the frequency.

POWer
Controls the output level.

Example: //Configure the second parameter with start, stop and stepsize values

```
CONF:PSW:Y:STAT ON
CONF:PSW:Y:SETT FREQ
CONF:PSW:Y:STAR 10MHZ
CONF:PSW:Y:STOP 100MHZ
CONF:PSW:Y:STEP 1MHZ
```

CONFigure:PSWeep:Y:STARt <Start>

This command defines the start value for the second parameter controlled by the parameter sweep.

Parameters:

<Start> <numeric value> whose unit depends on the parameter type you have selected with `CONFigure:PSWeep:Y:SETTing`:
- Hz in case of the center frequency
- dBm in case of the output level
- s in case of the delay between envelope and RF signal
- V in case of the envelope bias

Example: See `CONFigure:PSWeep:Y:SETTing`.

CONFigure:PSWeep:Y:STATe <State>

This command turns the second parameter controlled by the parameter sweep on and off.

Parameters:

<State> ON | OFF | 1 | 0

*RST: 1

Example: See `CONFigure:PSWeep:Y:SETTing`.

CONFigure:PSWeep:Y:STEP <Step>

This command defines the stepsize for the second parameter controlled by the parameter sweep.

Parameters:

<Step> <numeric value> whose unit depends on the parameter type you have selected with `CONFigure:PSWeep:Y:SETTing`:
- Hz in case of the center frequency
- dB in case of the output level
- s in case of the delay between envelope and RF signal
- V in case of the envelope bias

Example: See `CONFigure:PSWeep:Y:SETTing`.

CONFigure:PSWeep:Y:STOP <Stop>

This command defines the stop value for the second parameter controlled by the parameter sweep.

Parameters:

<Stop> <numeric value> whose unit depends on the parameter type you have selected with CONFigure:PSWeep:Y:SETTing:
• Hz in case of the center frequency
• dBm in case of the output level
• s in case of the delay between envelope and RF signal
• V in case of the envelope bias

Example: See CONFigure:PSWeep:Y:SETTing.

6.7 Analyzing Results

6.7.1 Configuring Traces

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:MODE <Trace>

This command selects the traces to be displayed in the graphical result displays.

Suffix:

<n> Window

<w>	irrelevant
<t>	Trace

Parameters:

<Trace> Available traces depend on the result display.

AVERage
The average is formed over several measurements.

BLANk
Removes the selected trace from the display.

MAXHold
The maximum value is determined over several measurements and displayed. The R&S FPS saves each trace point in the trace memory only if the new value is greater than the previous one.

MINHold
The minimum value is determined from several measurements and displayed. The R&S FPS saves each trace point in the trace memory only if the new value is lower than the previous one.

VIEW
The current contents of the trace memory are frozen and displayed.

WRITe
Overwrite mode (default): the trace is overwritten by each measurement.

Example: `DISP:WIND1:TRAC1:MODE WRIT`

Manual operation: See "Trace Mode" on page 93

FORMat:DEXPort:DSEParator <Separator>

This command selects the decimal separator for data exported in ASCII format.

Parameters:

<Separator> POINt | COMMa

COMMa
Uses a comma as decimal separator, e.g. *4,05*.

POINt
Uses a point as decimal separator, e.g. *4.05*.

*RST: *RST has no effect on the decimal separator.
Default is POINt.

Example: `FORM:DEXP:DSEP POIN`
Sets the decimal point as separator.

Manual operation: See " Decimal Separator " on page 95

FORMat:DEXPort:HEADer <State>

If enabled, additional instrument and measurement settings are included in the header of the export file for result data. If disabled, only the pure result data from the selected traces and tables is exported.

Parameters:

<State> ON | OFF | 0 | 1

*RST: 1

Manual operation: See " Include Instrument & Measurement Settings " on page 95

FORMat:DEXPort:TRACes <TracesToExport>

This command selects the data to be included in a data export file.

Setting parameters:

<TracesToExport>

SINGle
Exports a a single trace only.

ALL
Exports all traces in all windows in the current application.

*RST: SINGle

Example: //Export all traces

```
FORM:DEXP:TRAC ALL
```

MMEMory:STORe<n>:TRACe <Trace>, <FileName>

This command exports trace data to a file.

Suffix:

<n> 1..n
Window

Setting parameters:

<Trace> Number of the trace you want to save.
Note that the available number of traces depends on the selected result display. The value "0" exports all traces in a window.
To export all traces in all windows, turn on the feature to export all traces and all results first (`FORMat:DEXPort:TRACes`). The suffix at `STORe<n>` is ignored in that case.

Range: 0 to 6

<FileName> String containing the path and file name.

Example: //Export all traces in all windows to the specified directory.
`FORM:DEXP:TRAC ALL`
`MMEM:STOR:TRAC 0,'C:\AmplifierTrace'`
//Export all traces in window 2 to the specified directory.
`MMEM:STOR2:TRAC 0,'C:\AmplifierTrace'`
//Export the second trace in window 2 to the specified directory.
`MMEM:STOR2:TRAC 2,'C:\AmplifierTrace'`

Usage: Setting only

Manual operation: See "Selecting data to export" on page 94
See "Export Trace" on page 95

[SENSe:]DETector<t>:DEFault[:FUNCtion] <State>

Selects the default detector for result displays.

Suffix:
<t> Trace

Parameters:
<State> AVERage | OFF

Manual operation: See "Default Detector" on page 96

[SENSe:]DETector<t>:TRACe[:POINt] <Points>

Sets the maximum number of trace points to be used by detectors.

Suffix:
<t> Trace

Parameters:
<Points> numeric value

Manual operation: See "Max. Trace Points" on page 96

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:RESult <Trace>

Sets and queries the trace result type for the selected result display.

Suffix:
<n> Window

<w> irrelevant

<t> Trace

Parameters:
<Trace> BBI | BBPower | BBQ | RF | MEAS | MODel | REFerence

Example: `DISP:WIND:TRAC:RES MEAS`

Manual operation: See "Result Type" on page 93

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:PRESet <ResultType>

Applies predefined, commonly required trace settings to the selected window.

Suffix:

<n>	1..n Window
<w>	1..n subwindow
<t>	1..n Trace

Parameters:

<ResultType>	**ALL** Preset All Traces **MAM** Max | Avg | Min **MCM** Max | ClrWrite | Min
Example:	`DISP:WIND3:TRAC:PRES MCM` In window 3, the traces are set to the following modes: Trace 1: Max Hold Trace 2: Clear Write Trace 3: Min Hold

[SENSe:][WINDow<n>:]DETector<t>[:FUNCtion] <Detector>

Sets and queries the detector for the selected result display.

Suffix:

<n>	Window
<t>	Trace

Parameters:

<Detector>	NEGative | POSitive | NONE | AVERage
Manual operation:	See "Detector" on page 93

TRACe:IQ:DATA:FORMat <Slope>

Defines the I/Q data format.

Parameters:

<Slope>	COMPatible | IQBLock | IQPair
Example:	`TRACe1:IQ:DATA:FORMat IQP`

TRACe:IQ:DATA:MEMory? [<OffsetSamples>, <NoOfSamples>]

Tansfers raw I/Q data.

Query parameters:

<OffsetSamples> numeric value

<NoOfSamples> numeric value

Example: `TRACe1:IQ:DATA:MEMory?`

Usage: Query only

TRACe:IQ:RLENgth?

Returns the sweep length or capture length.

Return values:

<Samples> numeric value

Example: `TRACe:IQ:RLENgth?`

Usage: Query only

TRACe:IQ:SYNC:RLENgth?

Returns the sweep length or capture length at the current sample rate.

Return values:

<Samples> numeric value

Example: `TRACe:IQ:SYNC:RLENgth?`

Usage: Query only

6.7.2 Using Markers

6.7.2.1 General Marker Settings

CALCulate<n>:MARKer<m>:LINK <State>

This command turns marker coupling across result displays on and off.

Suffix:

<n> irrelevant

<m> irrelevant

Parameters:
<State> ON | OFF | 1 | 0
*RST: OFF

Example: //Couple markers
`CALC:MARK:LINK ON`

Manual operation: See "Link Markers Across Windows" on page 97

DISPlay[:WINDow<n>]:MINFo[:STATe] <State>

This command turns the marker information in all diagrams on and off.

Suffix:
<n> irrelevant

Parameters:
<State>

ON | 1
Displays the marker information in the diagrams.

OFF | 0
Hides the marker information in the diagrams.

*RST: 1

Example: `DISP:MINF OFF`
Hides the marker information.

Manual operation: See " Marker Info " on page 96

DISPlay[:WINDow<n>]:MTABle <DisplayMode>

This command turns the marker table on and off.

Suffix:
<n> irrelevant

Parameters:
<DisplayMode>

ON | 1
Turns on the marker table.

OFF | 0
Turns off the marker table.

AUTO
Turns on the marker table if 3 or more markers are active.

*RST: AUTO

Example: `DISP:MTAB ON`
Activates the marker table.

Manual operation: See " Marker Table Display " on page 96

6.7.2.2 Configuring Individual Markers

CALCulate<n>:DELTamarker<m>:AOFF

This command turns off *all* delta markers.

Suffix:

<n> Window

<m> irrelevant

Example: `CALC:DELT:AOFF`
Turns off all delta markers.

CALCulate<n>:DELTamarker<m>:LINK <State>

This command links delta marker <m> to marker 1.

If you change the horizontal position (x-value) of marker 1, delta marker <m> changes its horizontal position to the same value.

Suffix:

<n> Window

<m> Marker

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

Example: `CALC:DELT2:LINK ON`

Manual operation: See " Linking to Another Marker " on page 99

CALCulate<n>:DELTamarker<ms>:LINK:TO:MARKer<md> <State>

This command links delta marker <m1> to any active normal marker <m2>.

If you change the horizontal position of marker <m2>, delta marker <m1> changes its horizontal position to the same value.

Suffix:

<n> Window

<ms> source marker, see Marker

<md> destination marker, see Marker

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

Example: `CALC:DELT4:LINK:TO:MARK2 ON`
Links the delta marker 4 to the marker 2.

Manual operation: See " Linking to Another Marker " on page 99

CALCulate<n>:DELTamarker<m>:MREFerence <Reference>

This command selects a reference marker for a delta marker other than marker 1.

Suffix:

<n> Window

<m> Marker

Parameters:

<Reference>

Example: `CALC:DELT3:MREF 2`
Specifies that the values of delta marker 3 are relative to marker 2.

Manual operation: See " Reference Marker " on page 98

CALCulate<n>:DELTamarker<m>[:STATe] <State>

This command turns delta markers on and off.

If necessary, the command activates the delta marker first.

No suffix at DELTamarker turns on delta marker 1.

Suffix:

<n> Window

<m> Marker

Parameters:
<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

Example: `CALC:DELT2 ON`
Turns on delta marker 2.

Manual operation: See " Marker State " on page 98
See " Marker Type " on page 98

CALCulate<n>:DELTamarker<m>:TRACe <Trace>

This command selects the trace a delta marker is positioned on.

Note that the corresponding trace must have a trace mode other than "Blank".

If necessary, the command activates the marker first.

Suffix:
<n> Window

<m> Marker

Parameters:
<Trace> Trace number the marker is assigned to.

Example: `CALC:DELT2:TRAC 2`
Positions delta marker 2 on trace 2.

CALCulate<n>:DELTamarker<m>:X <Position>

This command moves a delta marker to a particular coordinate on the x-axis.

If necessary, the command activates the delta marker and positions a reference marker to the peak power.

Suffix:
<n> Window

<m> Marker

Parameters:
<Position> Numeric value that defines the marker position on the x-axis.

Range: The value range and unit depend on the measurement and scale of the x-axis.

Example: `CALC:DELT:X?`
Outputs the absolute x-value of delta marker 1.

Manual operation: See "Marker Position X-value " on page 98

CALCulate<n>:DELTamarker<m>:Y?

Queries the result at the position of the specified delta marker.

Suffix:

<n> 1..n

<m> 1..n

Return values:

<Result> Result at the position of the delta marker.
The unit is variable and depends on the one you have currently set.
Default unit: DBM

Usage: Query only

CALCulate<n>:MARKer<m>:AOFF

This command turns off all markers.

Suffix:

<n> Window

<m> Marker

Example: `CALC:MARK:AOFF`
Switches off all markers.

Manual operation: See " All Markers Off " on page 99

CALCulate<n>:MARKer<ms>:LINK:TO:MARKer<md> <State>

This command links normal marker <m1> to any active normal marker <m2>.

If you change the horizontal position of marker <m2>, marker <m1> changes its horizontal position to the same value.

Suffix:

<n> Window

<ms> source marker, see Marker

<md> destination marker, see Marker

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

Example: `CALC:MARK4:LINK:TO:MARK2 ON`
Links marker 4 to marker 2.

Manual operation: See " Linking to Another Marker " on page 99

CALCulate<n>:MARKer<m>[:STATe] <State>

This command turns markers on and off. If the corresponding marker number is currently active as a delta marker, it is turned into a normal marker.

Suffix:

<n> — Window

<m> — Marker

Parameters:

<State> — ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

Example: `CALC:MARK3 ON`
Switches on marker 3.

Manual operation: See " Marker State " on page 98
See " Marker Type " on page 98

CALCulate<n>:MARKer<m>:TRACe <Trace>

This command selects the trace the marker is positioned on.

Note that the corresponding trace must have a trace mode other than "Blank".

If necessary, the command activates the marker first.

Suffix:

<n> — Window

<m> — Marker

Parameters:

<Trace> — **1 to 4**
Trace number the marker is assigned to.

Example: //Assign marker to trace 1
`CALC:MARK3:TRAC 2`

Manual operation: See " Assigning the Marker to a Trace " on page 99

CALCulate<n>:MARKer<m>:X <Position>

This command moves a marker to a specific coordinate on the x-axis.

If necessary, the command activates the marker.

If the marker has been used as a delta marker, the command turns it into a normal marker.

Suffix:

<n> — Window

<m> Marker

Parameters:
<Position> Numeric value that defines the marker position on the x-axis.
The unit depends on the result display.
Range: The range depends on the current x-axis range.
Default unit: Hz

Example: `CALC:MARK2:X 1.7MHz`
Positions marker 2 to frequency 1.7 MHz.

Manual operation: See "Marker Position X-value " on page 98

CALCulate<n>:MARKer<m>:Y?

Queries the result at the position of the specified marker.

Suffix:
<n> 1..n

<m> 1..n

Return values:
<Result> Default unit: DBM

Usage: Query only

6.7.2.3 Positioning Markers

CALCulate<n>:DELTamarker<m>:MAXimum:LEFT

This command moves a delta marker to the next higher value.

The search includes only measurement values to the left of the current marker position.

Suffix:
<n> Window

<m> Marker

Manual operation: See " Search Next Peak " on page 100

CALCulate<n>:DELTamarker<m>:MAXimum:NEXT

This command moves a marker to the next higher value.

Suffix:
<n> 1..n
Window

<m> 1..n
Marker

Manual operation: See " Search Next Peak " on page 100

CALCulate<n>:DELTamarker<m>:MAXimum[:PEAK]

This command moves a delta marker to the highest level.

If the marker is not yet active, the command first activates the marker.

Suffix:
<n> Window

<m> Marker

Manual operation: See " Peak Search " on page 100

CALCulate<n>:DELTamarker<m>:MAXimum:RIGHt

This command moves a delta marker to the next higher value.

The search includes only measurement values to the right of the current marker position.

Suffix:
<n> Window

<m> Marker

Manual operation: See " Search Next Peak " on page 100

CALCulate<n>:DELTamarker<m>:MINimum:LEFT

This command moves a delta marker to the next higher minimum value.

The search includes only measurement values to the right of the current marker position.

Suffix:
<n> Window

<m> Marker

Manual operation: See " Search Next Minimum " on page 100

CALCulate<n>:DELTamarker<m>:MINimum:NEXT

This command moves a marker to the next higher minimum value.

Suffix:
<n> Window

<m> Marker

Manual operation: See " Search Next Minimum " on page 100

CALCulate<n>:DELTamarker<m>:MINimum[:PEAK]

This command moves a delta marker to the minimum level.

If the marker is not yet active, the command first activates the marker.

Suffix:
<n> Window

<m> Marker

Manual operation: See " Search Minimum " on page 100

CALCulate<n>:DELTamarker<m>:MINimum:RIGHt

This command moves a delta marker to the next higher minimum value.

The search includes only measurement values to the right of the current marker position.

Suffix:
<n> Window

<m> Marker

Manual operation: See " Search Next Minimum " on page 100

CALCulate<n>:MARKer<m>:MAXimum:LEFT

This command moves a marker to the next lower peak.

The search includes only measurement values to the left of the current marker position.

Suffix:
<n> Window

<m> Marker

Manual operation: See " Search Next Peak " on page 100

CALCulate<n>:MARKer<m>:MAXimum:NEXT

This command moves a marker to the next lower peak.

Suffix:

<n> Window

<m> Marker

Manual operation: See " Search Next Peak " on page 100

CALCulate<n>:MARKer<m>:MAXimum[:PEAK]

This command moves a marker to the highest level.

If the marker is not yet active, the command first activates the marker.

Suffix:

<n> Window

<m> Marker

Manual operation: See " Peak Search " on page 100

CALCulate<n>:MARKer<m>:MAXimum:RIGHt

This command moves a marker to the next lower peak.

The search includes only measurement values to the right of the current marker position.

Suffix:

<n> Window

<m> Marker

Manual operation: See " Search Next Peak " on page 100

CALCulate<n>:MARKer<m>:MINimum:LEFT

This command moves a marker to the next minimum value.

The search includes only measurement values to the right of the current marker position.

Suffix:

<n> Window

<m> Marker

Manual operation: See " Search Next Minimum " on page 100

CALCulate<n>:MARKer<m>:MINimum:NEXT

This command moves a marker to the next minimum value.

Suffix:

<n> Window

<m> Marker

Manual operation: See " Search Next Minimum " on page 100

CALCulate<n>:MARKer<m>:MINimum[:PEAK]

This command moves a marker to the minimum level.

If the marker is not yet active, the command first activates the marker.

Suffix:

<n> Window

<m> Marker

Manual operation: See " Search Minimum " on page 100

CALCulate<n>:MARKer<m>:MINimum:RIGHt

This command moves a marker to the next minimum value.

The search includes only measurement values to the right of the current marker position.

Suffix:

<n> Window

<m> Marker

Manual operation: See " Search Next Minimum " on page 100

6.7.3 Configuring Numerical Result Displays

DISPlay[:WINDow<n>]:PTABle:ITEM <Item>, <State>
DISPlay[:WINDow<n>]:PTABle:ITEM? <Item>

This command adds and removes results from the Parameter Sweep Table.

Suffix:
<n> 1..n
Window
Note that you have to include the `WINDow` syntax element if the Parameter Sweep Table is in a window other than window 1.

Parameters:
<State> ON | OFF | 1 | 0
*RST: All results are 1.

Parameters for setting and query:
<Item> Selects the result.
See the table at `CONFigure:PSWeep:Z<n>:RESult` for a list of available parameters.

Example: `DISP:PTAB:ITEM RMS,OFF`
Removes the RMS Power result from the Parameter Sweep Table.

DISPlay[:WINDow<n>]:PTABle:ITEM:ALL <State>

This command adds and removes all parameter sweep results from the parameter sweep table.

Suffix:
<n> Window

Setting parameters:
<State> ON | OFF | 1 | 0

Example: //Display all parameter sweep results
`DISP:PTAB:ITEM:ALL ON`

Usage: Setting only

DISPlay[:WINDow<n>]:TABLe:ITEM <Item>, <State>
DISPlay[:WINDow<n>]:TABLe:ITEM? <Item>

This command adds and removes results from the result summary.

Suffix:
<n> Window
Note that you have to include the `WINDow` syntax element if the result summary is in a window other than window 1.

Parameters:
<State> ON | OFF | 1 | 0
*RST: All results are 1.

Parameters for setting and query:
<Item> Selects the result.
See the table below for a list of available parameters.

Example: //Removes the gain imbalance result from the result summary.
`DISP:TABL:ITEM GIMB,OFF`
//Query if frequency error result is calculated
`DISP:WIND2:TABL:ITEM? FERR`
would return, e.g.
`1`

SCPI parameter	Result
AMWidth	AM curve width
CFIN	Crest factor in
CFOU	Crest factor out
FERRor	Frequency error
GAIN	Gain
GIMBalance	Gain Imbalance
IQIMbalance	I/Q imbalance
IQOFfset	I/Q offset
MERRor	Magnitude error
OUTP1db	1 dB output compression point
OUTP2db	2 dB output compression point
OUTP3db	3 dB output compression point
P1DB	1 dB compression point
P2DB	2 dB compression point
P3DB	3 dB compression point
PC	Average power consumption
PERRor	Phase error
PINPut	Power in
PMWidth	PM curve width
POUTput	Power out
QERRor	Quadrature error
REVM	Raw EVM
RMEVm	Raw model EVM
SERRor	Sample rate error

DISPlay[:WINDow<n>]:TABLe:ITEM:MACCuracy:ALL <State>

This command adds and removes all modulation accuracy results from the result summary.

Suffix:
<n> Window

Setting parameters:
<State> ON | OFF | 1 | 0

Example: //Display all modulation accuracy results
`DISP:TABL:ITEM:MACC:ALL ON`

Usage: Setting only

DISPlay[:WINDow<n>]:TABLe:ITEM:POWer:ALL <State>

This command adds and removes all power results from the result summary.

Suffix:
<n> Window

Setting parameters:
<State> ON | OFF | 1 | 0

Example: //Display all power result
`DISP:TABL:ITEM:POW:ALL ON`

Usage: Setting only

6.7.4 Configuring the Statistics Table

DISPlay[:WINDow<n>]:STABle:ITEM <Item>, <State>
DISPlay[:WINDow<n>]:STABle:ITEM? <Item>

This command adds and removes results from the statistics table.

Suffix:
<n> Window
Note that you have to include the `WINDow` syntax element if the statistics table is in a window other than window 1.

Parameters:
<State> ON | OFF | 1 | 0
*RST: All results are 1.

Parameters for setting and query:
<Item> Selects the result.
See the table in the desription of `DISPlay[:WINDow<n>]:TABLe:ITEM` on page 297 for a list of available parameters.

Example: //Removes the gain imbalance result from the statistics table.
`DISP:STAB:ITEM GIMB,OFF`
//Query if frequency error result is calculated
`DISP:WIND2:STAB:ITEM? FERR`
would return, e.g.
`1`

DISPlay[:WINDow<n>]:STABle:ITEM:MACCuracy:ALL <State>

This command adds and removes all modulation accuracy results from the statistics table.

Suffix:
<n> Window

Setting parameters:
<State> ON | OFF | 1 | 0

Example: //Display all modulation accuracy results
`DISP:STAB:ITEM:MACC:ALL ON`

Usage: Setting only

DISPlay[:WINDow<n>]:STABle:ITEM:POWer:ALL <State>

This command adds and removes all power results from the statistics table.

Suffix:
<n> Window

Setting parameters:
<State> ON | OFF | 1 | 0

Example: //Display all power result
`DISP:STAB:ITEM:POW:ALL ON`

Usage: Setting only

6.7.5 Configuring Result Display Characteristics

CALCulate<n>:AMPM:DEFinition <ResultType>

This command selects the way the AM/PM results are calculated.

Suffix:
<n> irrelevant

Parameters:
<ResultType> **MREF**
Subtracts the reference trace from the measurement trace.
This is the inverse of the default `REAFMeas` method.

REFMeas
Subtracts the measurement trace from the reference trace.

*RST: REFMeas

Example: `CALC:AMPM:DEF?`
would return, e.g.
`REFM`

Manual operation: See "Selecting the AM/PM definition" on page 103

CALCulate<n>:PREFerence:X <ResultType>

This command selects the type of information displayed on x-axis in the following result displays.

- EVM vs Power
- AM/PM
- Gain Compression

Suffix:
<n> Window

Parameters:
<ResultType> **PINPut**
Shows the corresponding result against the input level.

POUTput
Shows the corresponding result against the output level.

Example: //Displays the result (for example AM/PM) against the input level.
`CALC:GAIN:X PINP`

Manual operation: See "Selecting the Power Reference" on page 103

CALCulate<n>:UNIT:ANGLe <Unit>

This command selects the unit for results that display the phase.

Suffix:
<n> Window

Parameters:
<Unit> **DEG**
Phase displayed in degrees.

RAD
Phase displayed in radians.

Example: //Show the phase results in degrees
`CALC:UNIT:ANGL DEG`

Manual operation: See "Selecting the unit for phase results" on page 103

CONFigure:PSWeep:Z<n>:RESult <Result>

This command selects the result type displayed on the z-axis of the parameter sweep diagram.

Suffix:
<n> 1..n
Window

Parameters:
<Result> See table below for supported result types.

Example: `CONF:PSW:Z:RES EVM`
Displays the EVM against two parameters in the Parameter Sweep result display.

Manual operation: See "Selecting the result type displayed in the parameter sweep diagram" on page 103

ACBM	ACLR Balanced Magnitude
ACL1	ACLR Adjacent 1 Lower
ACP	Adjacent Channel Power
ACU1	ACLR Adjacent 1 Upper
AMWidth	AM/AM Curve Width
CFACtor	Crest Factor
EVM	EVM
GAIN	Gain
PMWidth	AM/PM Curve Width
RMS	RMS Power

DISPlay[:WINDow<n>]:TDOMain:X[:SCALe]:DURation <Time>

This command defines the amount of data displayed on the x-axis of the time domain result display.

Prerequisites for this command

- Turn off automatic scaling (`DISPlay[:WINDow<n>]:TDOMain:X[:SCALe]:MODE`).

Suffix:
<n> Window

Parameters:
<Time> <numeric value>

Time that is displayed on the x-axis, beginning at the offset defined with DISPlay[:WINDow<n>]:TDOMain:X[:SCALe]:OFFSet.

Default unit: s

Example: //Scale the x-axis in the time domain result display
```
DISP:TDOM:X:MODE OFF
DISP:TDOM:X:DUR 12us
```

Manual operation: See "Configuring the time domain result display" on page 102

DISPlay[:WINDow<n>]:TDOMain:X[:SCALe]:MODE <State>

This command turns automatic scaling of the x-axis in the time domain result display on and off.

Suffix:
<n> Window

Parameters:
<State> **ON | 1**
Turns on automatic scaling of the x-axis.

OFF | 0
Turns on manual scaling of the x-axis.

Example: //Turn on manual scaling of the x-axis
```
DISP:TDOM:X:MODE OFF
```

Manual operation: See "Configuring the time domain result display" on page 102

DISPlay[:WINDow<n>]:TDOMain:X[:SCALe]:OFFSet <Time>

This command defines the origin of the x-axis in the time domain result display.

Prerequisites for this command

- Turn off automatic scaling (DISPlay[:WINDow<n>]:TDOMain:X[:SCALe]:MODE).

Suffix:
<n> Window

Parameters:
<Time> <numeric value>

Time offset relative to the first recorded sample (when synchronization has failed) or the first sample of the synchronized data (when synchronization was successful).

Default unit: s

Example: Defines an offset
`DISP:TDOM:X:MODE OFF`
`DISP:TDOM:X:OFFS 12us`

Manual operation: See "Configuring the time domain result display" on page 102

DISPlay[:WINDow<n>]:TDOMain:Y[:SCALe]:NORMalise[:STATe] <State>

This command turns normalization of the results in the time domain result display on and off.

Suffix:
<n> Window

Parameters:
<State> ON | OFF | 1 | 0

Example: //Normalize the results in the time domain result display to 1
`DISP:TDOM:Y:NORM ON`

Manual operation: See "Configuring the time domain result display" on page 102

6.7.6 Scaling the Diagram Axes

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:X[:SCALe]:AUTO <State>

This command turns automatic scaling of the x-axis in graphical result displays on and off.

Suffix:
<n> Window

<w> irrelevant

<t> irrelevant

Parameters:
<State> **OFF | 0**
Selects manual scaling of the diagram.

ON | 1
Automatically scales the diagram when new results are available.

ONCE
Automatically scales the diagram once whenever required.

*RST: ON

Example: //Scale the axis each time new results are available
`DISP:TRAC:X:AUTO ON`

Manual operation: See "Scaling the x-axis automatically" on page 104

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:X[:SCALe]:MAXimum <Value>

This command defines the value at the top of the x-axis.

Suffix:

<n> Window

<w> irrelevant

<t> irrelevant

Parameters:

<Value> <numeric value>

Default unit: Depends on the result display.

Example: //Define x-axis level range
```
DISP:TRAC:x:AUTO OFF
DISP:TRAC:x:MIN -10DBM
DISP:TRAC:x:MAX -110DBM
```

Manual operation: See "Scaling the x-axis manually" on page 105

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:X[:SCALe]:MINimum <Value>

This command defines the value at the bottom of the y-axis.

Suffix:

<n> Window

<w> irrelevant

<t> irrelevant

Parameters:

<Value> <numeric value>

Default unit: Depends on the result display.

Example: //Define x-axis level range
DISP:TRAC:X:AUTO OFF
DISP:TRAC:X:MIN -10DBM
DISP:TRAC:X:MAX -110DBM

Manual operation: See "Scaling the x-axis manually" on page 105

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:X[:SCALe]:PDIVision <Distance>

This command defines the distance between the horizontal grid lines in graphical result displays.

Prerequisites for this command

- Turn off automatic scaling (DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:X[:SCALe]:AUTO).

Suffix:

<n>	Window
<w>	irrelevant
<t>	irrelevant

Parameters:

<Distance> <numeric value>

Default unit: Depends on the result display.

Example: //Define a distance of 5 dBm between the grid lines
DISP:TRAC:X:SCAL:AUTO OFF
DISP:TRAC:X:PDIV 5DBM

Manual operation: See "Scaling the x-axis manually" on page 105

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:X[:SCALe]:UNIT?

This command queries the unit of the x-axis

Suffix:

<n>	Window
<w>	irrelevant
<t>	irrelevant

Return values:

<Unit> Unit of the x-axis in the selected window.

Example: DISP:WIND4:TRAC:X:UNIT?
would return, e.g.
SEC

Usage: Query only

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:AUTO <State>

This command turns automatic scaling of the y-axis in graphical result displays on and off.

Suffix:

<n>	Window
<w>	irrelevant
<t>	irrelevant

Parameters:

<State>

OFF
Selects manual scaling of the diagram.

ON
Automatically scales the diagram when new results are available.

ONCE
Automatically scales the diagram once whenever required.

*RST: ON

Example:
//Scale the axis each time new results are available
```
DISP:TRAC:Y:AUTO ON
```

Manual operation: See "Scaling the y-axis automatically" on page 106

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:MAXimum <Value>

This command defines the value at the top of the y-axis.

Suffix:

<n>	Window
<w>	irrelevant
<t>	irrelevant

Parameters:

<Value> <numeric value>

Default unit: Depends on the result display.

Example:
//Define y-axis level range
```
DISP:TRAC:Y:AUTO OFF
DISP:TRAC:Y:MIN -10DBM
DISP:TRAC:Y:MAX -110DBM
```

Manual operation: See "Scaling the y-axis manually" on page 106

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:MINimum <Value>

This command defines the value at the bottom of the y-axis.

Suffix:

<n>	Window
<w>	irrelevant
<t>	irrelevant

Parameters:

<Value> <numeric value>

Default unit: Depends on the result display.

Example:
//Define y-axis level range

```
DISP:TRAC:Y:AUTO OFF
DISP:TRAC:Y:MIN -10DBM
DISP:TRAC:Y:MAX -110DBM
```

Manual operation: See "Scaling the y-axis manually" on page 106

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:PDIVision <Distance>

This command defines the distance between the grid lines in graphical result displays.

Prerequisites for this command

- Turn off automatic scaling (DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:AUTO).

Suffix:

<n>	Window
<w>	irrelevant
<t>	irrelevant

Parameters:

<Distance> <numeric value>

Default unit: Depends on the result display.

Example:
//Define a distance of 5 dBm between the grid lines

```
DISP:TRAC:Y:SCAL:AUTO OFF
DISP:TRAC:Y:PDIV 5DBM
```

Manual operation: See "Scaling the y-axis manually" on page 106

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:RPOSition <Position>

This command defines the position of the reference value.

You can define the reference value with DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:RVALue.

Suffix:

<n> Window

<w> irrelevant

<t> irrelevant

Parameters:

<Position> <numeric value>

Default unit: %

Example: //Position the reference value at the 80 % mark of the y-axis

```
DISP:TRAC:Y:AUTO OFF
DISP:TRAC:Y:RVAL 0DBM
DISP:TRAC:Y:RPOS 80
```

Manual operation: See "Scaling the y-axis manually" on page 106

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:RVALue <Reference>

This command defines the reference value of a result display.

Prerequisites for this command

- Turn off automatic scaling (`DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:AUTO`).

Suffix:

<n> Window

<w> irrelevant

<t> irrelevant

Parameters:

<Reference> <numeric value>

Default unit: The unit depends on the result display.

Example: //Define a reference value of 10 dB

```
DISP:TRAC:Y:AUTO OFF
DISP:TRAC:Y:RVAL 10DB
```

Manual operation: See "Scaling the y-axis manually" on page 106

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:UNIT?

This command queries the unit of the y-axis

Suffix:

<n> Window

<w> irrelevant

<t> irrelevant

Return values:

<Unit> Unit of the y-axis in the selected window.

Example: `DISP:WIND3:TRAC:Y:UNIT?`
would return, e.g.
`DBM`

Usage: Query only

6.7.7 Managing Measurement Data

MMEMory:LOAD:IQ:STATe <1>, <FileName>

This command restores the currently captured I/Q data to a file.

After restoring the I/Q data, the application also analyzes the data again.

Setting parameters:
<1>

<FileName> String containing the path and file name.

Example: `MMEM:LOAD:IQ:STAT 1,'C:\IQData\Amplfier.iq.tar'`
Restores the specified I/Q data.

Usage: Setting only

MMEMory:STORe<n>:IQ:COMMent <Comment>

This command defines a comment for I/Q data you want to store.

Suffix:
<n> 1..n
irrelevant

Setting parameters:
<Comment> String containing the comment.

Example: See `MMEMory:STORe<n>:IQ:STATe`.

MMEMory:STORe<n>:IQ:STATe <Number>, <FileName>

This command stores the currently captured I/Q data to a file.

In secure user mode, settings that are stored on the instrument are stored to volatile memory, which is restricted to 256 MB. Thus, a "memory limit reached" error can occur although the hard disk indicates that storage space is still available.

To store data permanently, select an external storage location such as a USB memory device.

For details, see "Protecting Data Using the Secure User Mode" in the "Data Management" section of the R&S FPS base unit user manual.

Suffix:
<n> 1..n
irrelevant

Setting parameters:
<Number> Always '1'.

<FileName> String containing the path and file name.
The file type is .iq.tar.

Example:
```
MMEM:STOR:IQ:COMM 'A sensible comment'
MMEM:STOR:IQ:STAT 1,'C:\IQData\Amplfier.iq.tar'
```
Saves the I/Q data to the specified file and adds a sensible comment.

Usage: Setting only

6.8 Deprecated Remote Commands for Amplifier Measurements

Following is a list of deprecated remote commands. The remote commands are still supported to maintain compatibility to previous versions of amplifier measurements, but it is strongly recommended to use the command system in the way it is meant to be used in the latest version of the R&S FPS-K18.

Legacy command	Replaced by	Comment
CALCulate:GAIN:X	CALCulate:PREFerence:X	
CONFigure:DPD:MODorder		
CONFigure:MODeling:AMAM:MORDer	CONFigure:MODeling:AMAM:ORDer	
CONFigure:MODeling:AMPM:MORDer	CONFigure:MODeling:AMPM:ORDer	
CONFigure:MODeling:ORDer	CONFigure:MODeling:SEQuence	
CONFigure:PSERvoing:GATTenuation	CONFigure:PSERving:OPOWer	
FETCh:POWer:CURRent[:RESult]	FETCh:BBPower:CURRent[:RESult]	
FETCh:POWer:MAXimum[:RESult]	FETCh:BBPower:MAXimum[:RESult]	
FETCh:POWer:MINimum[:RESult]	FETCh:BBPower:MINimum[:RESult]	

List of commands

Index