R&S®FSW-K18
Power Amplifier and Envelope Tracking Measurements
User Manual

1176989302
Version 26

ROHDE&SCHWARZ
Make ideas real

This manual applies to the following FSW models with firmware version 6.10 and later:

- R&S®FSW8 (1331.5003K08 / 1312.8000K08)
- R&S®FSW13 (1331.5003K13 / 1312.8000K13)
- R&S®FSW26 (1331.5003K26 / 1312.8000K26)
- R&S®FSW43 (1331.5003K43 / 1312.8000K43)
- R&S®FSW50 (1331.5003K50 / 1312.8000K50)
- R&S®FSW67 (1331.5003K67 / 1312.8000K67)
- R&S®FSW85 (1331.5003K85 / 1312.8000K85)

The following firmware options are described:

- R&S®FSW-K18 (1325.2170.02)
- R&S®FSW-K18D (1331.6845.02)
- R&S®FSW-K18F (1338.7230.02)
- R&S®FSW-K18M (1345.1470.02)

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

1176.9893.02 | Version 26 | R&S®FSW-K18

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

Contents

1 Welcome to the amplifier measurement application

The R&S FSW-K18 is a firmware application that adds functionality to measure the efficiency of amplifiers with the FSW signal analyzer. You extend the amplifier application with the R&S FSW-K18D, which adds direct digital predistortion (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 FSW user manual. The latest versions of the manuals are available for download at the product homepage.

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

Find various videos on Rohde & Schwarz products and test and measurement topics on YouTube: https://www.youtube.com/@RohdeundSchwarz

Installation

Find detailed installing instructions in the getting started or the release notes of the FSW.

1.1 Starting the application

The amplifier measurement application adds a new type of measurement to the FSW.

To activate the amplifier application

1. Press [MODE] on the front panel of the FSW.

   A dialog box opens that contains all operating modes and applications currently available on your FSW.

2. Select the "Amplifier" item.

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

Sequencer

Note that you can use the Amplifier measurement application with the Sequencer that is available with the FSW. The functionality is the same as in the spectrum application. Refer to the FSW user manual for more information.

1.2 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 1-1: Screen layout of the amplifier measurement application

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

For a description of the elements not described below, refer to the getting started of the FSW.

Channel bar information

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

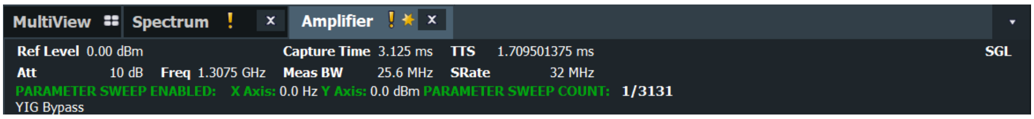


Figure 1-2: Channel bar of the amplifier application

"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 1-3: 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.

2 Measurements and result displays

During an amplifier measurement, I/Q data from the input signal is captured for a specified time or for a specified record length. In addition, a reference signal is provided that describes the characteristics of the input signal. The Amplifier measurement application synchronizes the measured signal and compares it with the ideal reference signal, and determines deviations in characteristic parameters. These parameters can either be displayed as traces, in a table, or be evaluated statistically over a series of measurements.

2.1 Evaluation methods for amplifier measurements

The data that was measured by the Amplifier measurement application can be evaluated using various different methods.

By default, the amplifier measurement results are displayed in the following windows:

- "Magnitude Capture" on page 22
- "Numeric Result Summary" on page 28
- "Spectrum FFT" on page 29
- "AM/AM" on page 14
- "Time Domain" on page 30
- "AM/PM (Input Power)" on page 15

The following evaluation methods are available for amplifier measurements:

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 FSW-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 3.17, "Adjacent channel leakage error (ACLR) measurements", on page 128.

Remote command:
Selection: `LAY:ADD? '1',LEFT,ACP`
Result query: `CALCulate<n>:MARKer<m>:FUNCtion:POWer:RESult?`
on page 300

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.

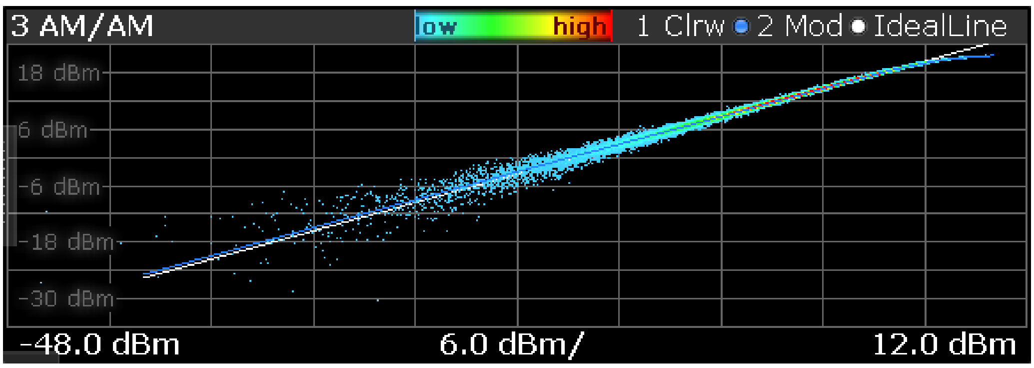
The width of the "AM/AM" trace is an indicator of memory effects: the larger the width of the trace, the more memory effects occur. The "AM/AM" Curve Width is shown in the numerical Result Summary.

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 348

AM/PM (Input Power)
The "AM/PM" result disyplay 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 is 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".

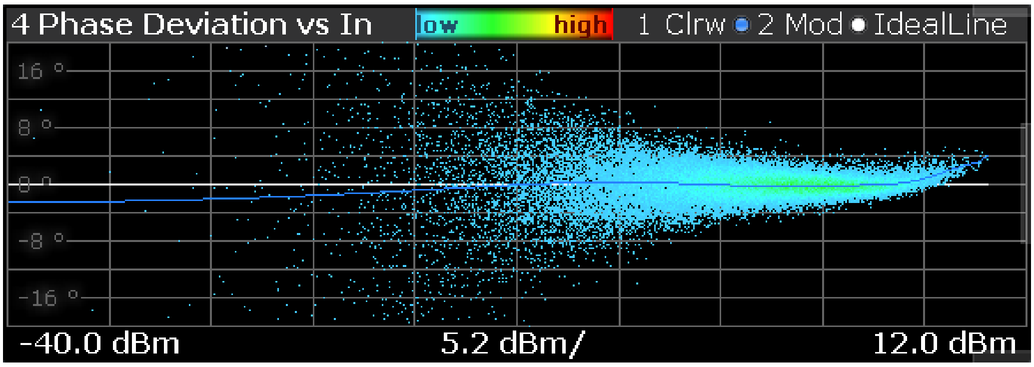
The x-axis shows the levels of all samples of the reference signal (input power) or the measurement signal (output power) in dBm. You can select the reference of the x-axis (input or output power) in the "Result Configuration" dialog box.

The y-axis shows the phase of the (unwrapped) 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.



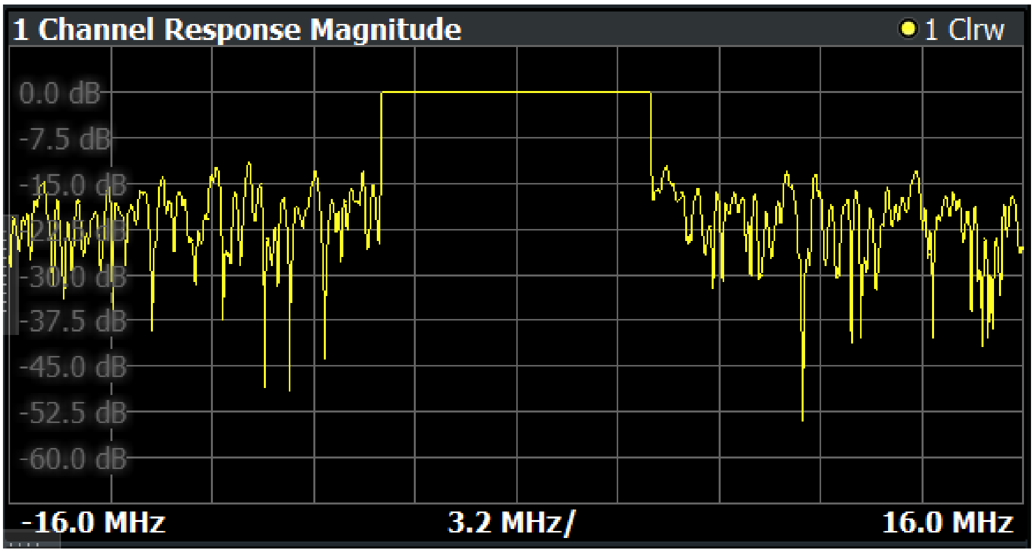
Remote command:
Selection: `LAY:ADD? '1',LEFT,AMPM`
Result query: `TRACe<n>[:DATA]?` on page 348

Channel Response Magnitude / Channel Response Phase / Group Delay (FSW-K18F)
The channel response and group delay result displays show the deviation of the measured signal compared to the reference signal within the measured channel. The result displays contain a single trace.

Outside of the occupied bandwidth, the reference signal values usually lie below the measured noise floor. This can result in large peaks on the trace in these areas (usually to the left and right of the channel). Note that because of the automatic y-axis scaling, the trace can appear in parts as a straight horizontal line. In that case, adjust the scale of the y-axis manually.

Channel Response Magnitude

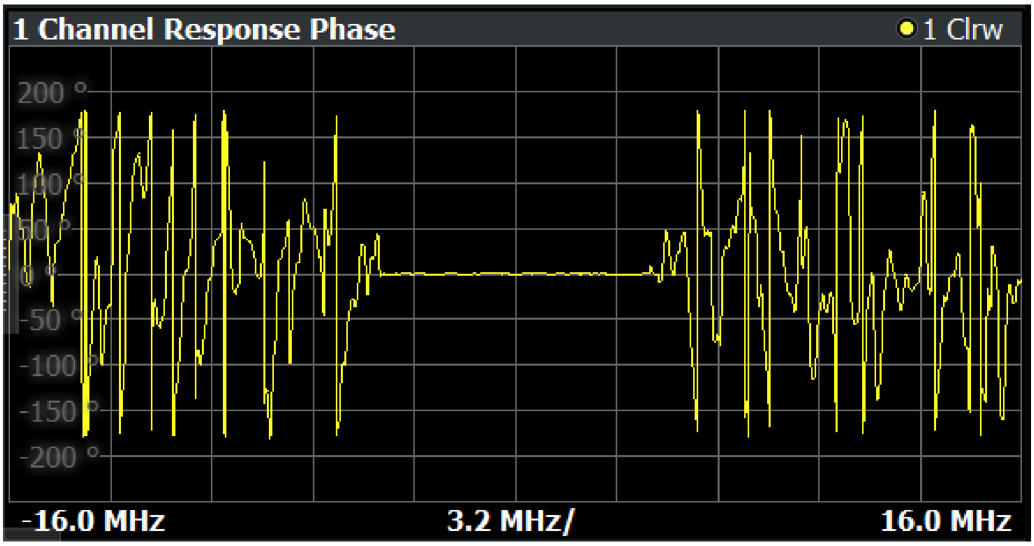
The "Channel Response Magnitude" result display analyzes the magnitude characteristics of the signal over the measurement bandwidth.



For the "Channel Response Magnitude", the y-axis shows the deviation of the measured magnitude relative to the transmitted signal power of the signal generator in dB. The x-axis shows the frequency over which the signal was measured.

Channel Response Phase

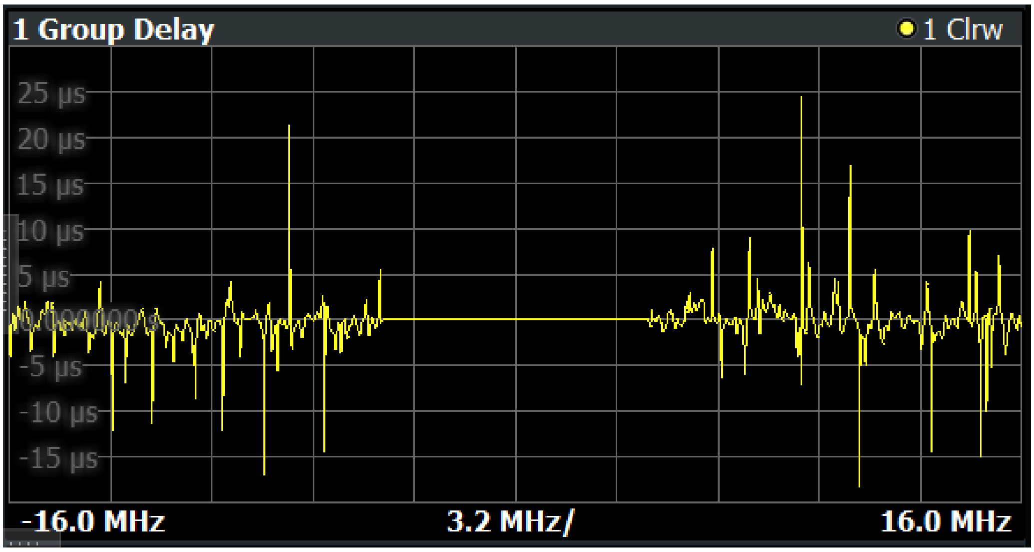
The "Channel Response Phase" result display analyzes the phase characteristics of the signal over the measurement bandwidth.



For the "Channel Response Phase", the y-axis shows the phase deviation relative to the reference signal. The unit depends on your selection. The x-axis shows the frequency over which the signal was measured.

Group Delay

The "Group Delay" result display analyzes the relative group delay of the signal over the measurement bandwidth.



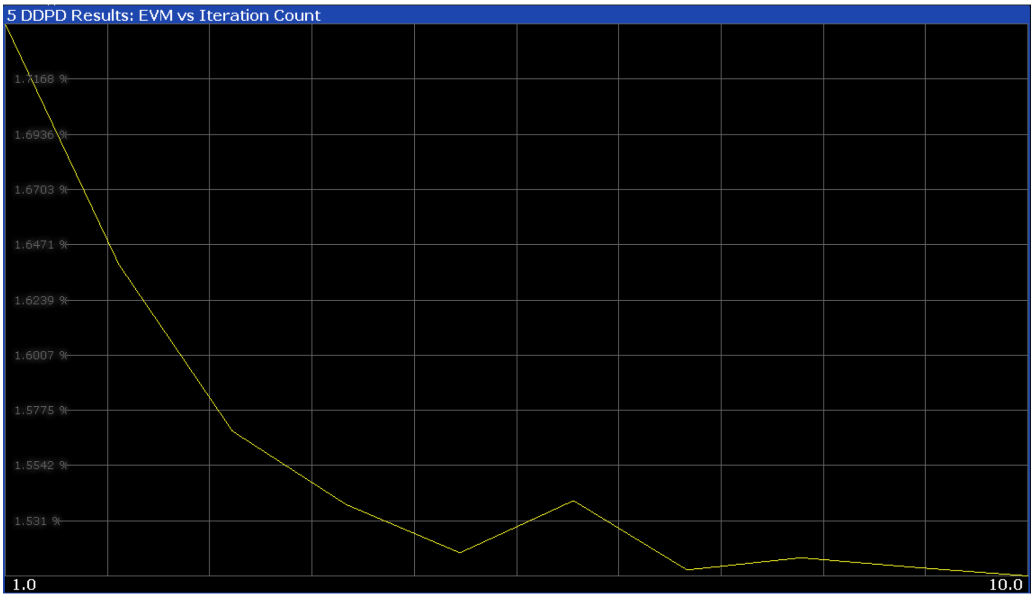
For the "Group Delay", the y-axis shows the measured time delay relative to the reference signal in seconds. The x-axis shows the frequency over which the signal was measured.

Remote command:
Selection (magnitude): `LAY:ADD? '1',LEFT,MRES`
Selection (phase): `LAY:ADD? '1',LEFT,PRES`
Selection (group delay): `LAY:ADD? '1',LEFT,GDEL`
Result query: `TRACe<n>[:DATA]?` on page 348

DDPD Results (FSW-K18D)

The "DDPD Results" result display shows a selectable result over all iterations of the direct DPD. This allows you to verify the direct DPD convergence, and to select the ideal iteration step for further processing (e.g. in FSW-K18M). It is only available with application FSW-K18D installed.

Select the result display before starting the direct DPD. Configure the result type in the Chapter 4.4, "Result display settings", on page 151.



The following result types are available:

"EVM"
Error vector magnitude
Remote command:
`CONFigure:DDPD:WINDow<n>:RESult EVM`

"ACLR Adj Upper"
Power of the upper adjacent channel
Remote command:
`CONFigure:DDPD:WINDow<n>:RESult ACU1`/

"ACLR Adj Lower"
Power of the lower adjacent channel
Remote command:
`CONFigure:DDPD:WINDow<n>:RESult ACL1`

"Bal ACLR Magnitude"
Difference between the lower and upper adjacent channel power
Remote command:
`CONFigure:DDPD:WINDow<n>:RESult ACB1`

Remote command:
Selection: `LAY:ADD? '1',LEFT,DDPD`
Result query: `TRACe<n>[:DATA]?` on page 348

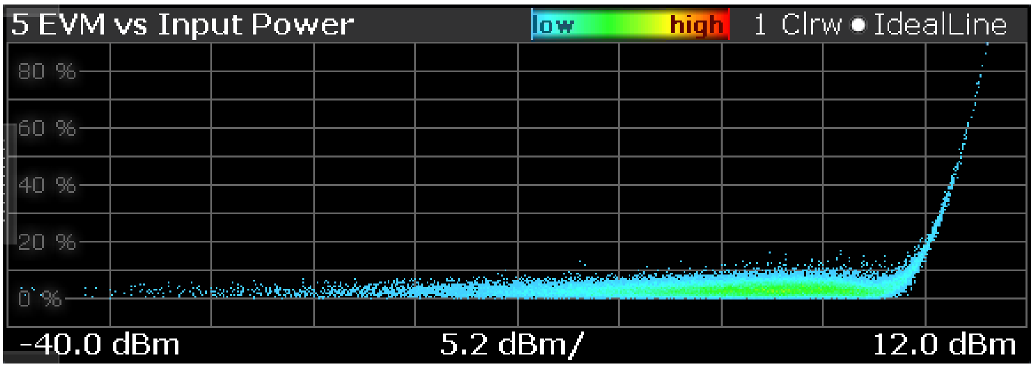
EVM vs Power
The "EVM vs Power" result display shows the EVM against the measured power values.

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 348

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

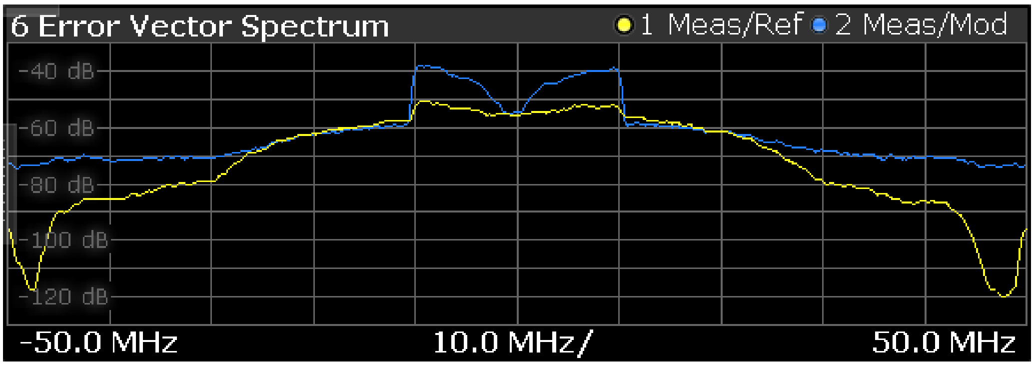
The EV is a measure of the modulation accuracy. It compares two signals and shows the distance of the measured constellation points and the ideal constellation points.

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 348

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.

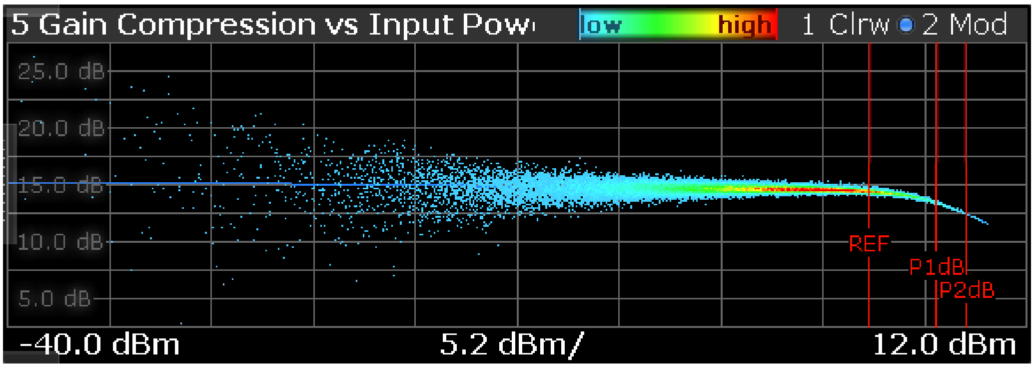
- Measured signal
  Shows the gain characteristics of the DUT.
  The software uses the reference signal in combination with the synchronized measurement signal to calculate a software model that describes the characteristics of the device under test.
  The measured gain is represented by a colored cloud 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 fre-

quencies, 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 348

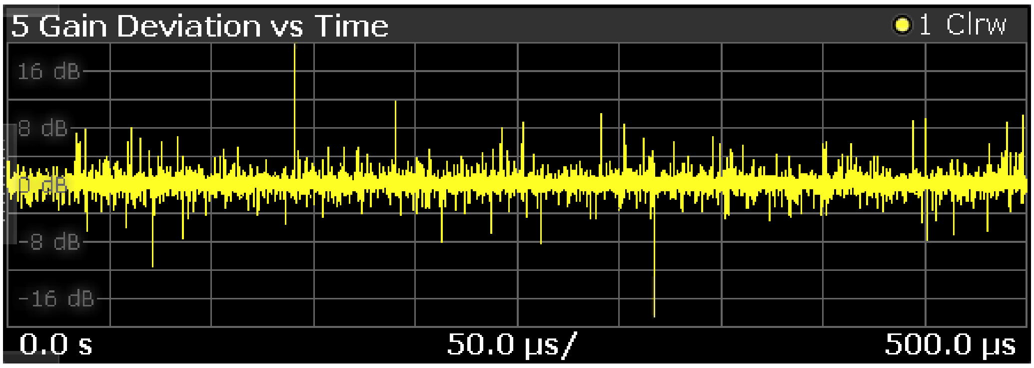
Gain Deviation vs Time

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 348

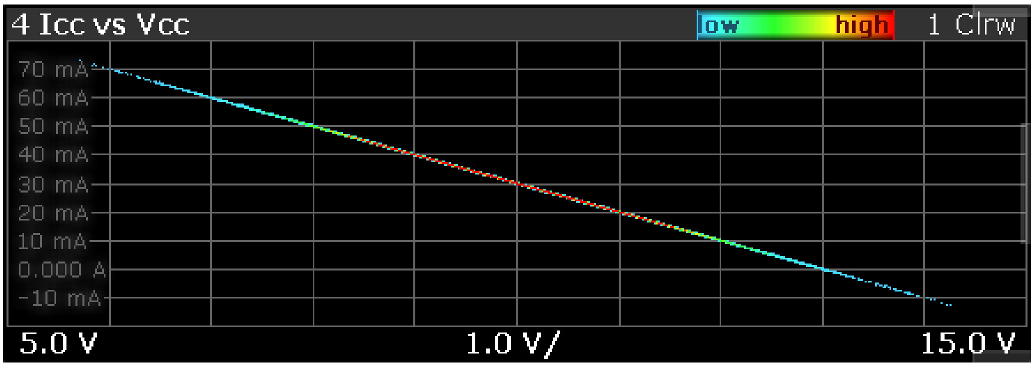
Vcc vs Icc

Note: This result display requires you to turn on the baseband input.

The "V_{cc} vs I_{cc}" result display shows the supply voltage that has been measured on baseband input Q against the current consumption that has been measured on baseband input I (using a shunt resistor or current probe).

The x-axis shows the voltage (V). The y-axis shows the current (A).

The resulting trace is usually represented by a 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 level / gain combination occurs. Blue pixels represent low statistical frequencies, red pixels high statistical frequencies. A color map is provided within the result display.



Remote command:
Selection: `LAY:ADD? '1',LEFT,VICC`
Result query: `TRACe<n>[:DATA]?` on page 348

Magnitude Capture (RF, I and Q)

Note: The magnitude capture I and Q results displays require you to turn on the baseband input.

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

It is available for the data recorded on the RF input and both baseband inputs (I and Q channels). (Note that the I and Q channel capture buffers are only available when parallel baseband capture is active.)

The capture buffer shows the signal level over time. The unit is either dBm (RF capture), V or A (baseband capture).

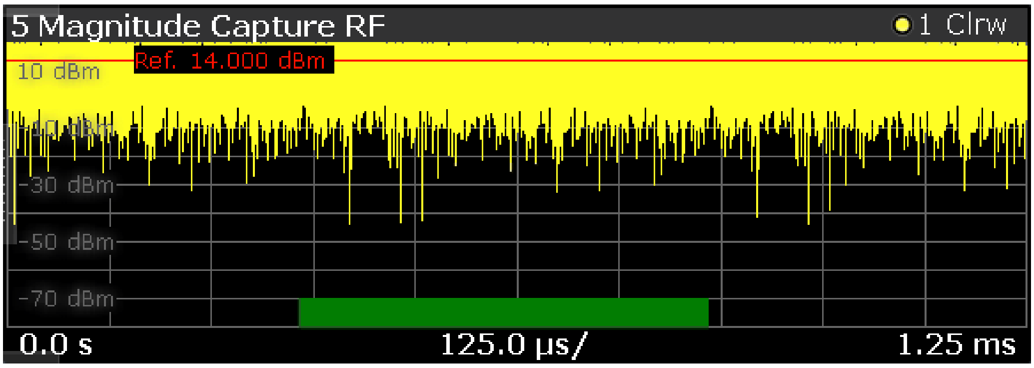
For a baseband capture, all multipliers and offsets are already included in the results.

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`
Selection (I): `LAY:ADD? '1',LEFT,IMAG`
Selection (Q): `LAY:ADD? '1',LEFT,QMAG`
Result query: `TRACe<n>[:DATA]?` on page 348

Memory DPD Coefficients

The "Memory DPD Coefficients" result table shows basically complex filter coefficients for each polynomial degree. The two lines "1(Real)" and "1(Imag)" describe the complex impulse response for polynomial degree 1 (linear) of a filter from left to right. It is only available with application FSW-K18M installed.

8 Memory DPD Coefficients

Polynomial Order	Memory Order					
	0	1	2	3	4	5
1 (Real)	1.60787	-8.17859	18.4234	-19.2749	11.1634	-2.71031
1 (Imag)	0.360876	-1.83454	3.68814	-3.7469	1.8992	-0.345397
2 (Real)	0.0355388	-0.1169	-0.494853	-0.145286	0.138638	-0.0629907
2 (Imag)	0.0050489	0.000310979	-0.0457932	0.01782	-0.00895232	0.0174621
3 (Real)	-0.128389	0.47902	2.59296	0.453788	-0.556218	0.250055
3 (Imag)	-0.0142906	-0.0080389	0.224799	-0.017381	0.0013791	-0.062957
4 (Real)	0.155902	-0.681237	-5.47602	-0.356508	0.775582	-0.360015
4 (Imag)	-0.0101379	0.113657	-0.88731	0.0592651	-0.0296721	0.109657
5 (Real)	-0.0531163	0.317786	4.07806	-0.0292011	-0.389324	0.182799
5 (Imag)	0.021386	-0.118354	0.868153	-0.041535	0.0369386	-0.0794943

Remote command:
Selection: `LAY:ADD? '1',LEFT,MDPD`
Result query: `FETCh:MDPD:COEFficients?` on page 292

PAE vs Input Power / PAE vs Output Power

Note: This result display requires you to turn on the baseband input.

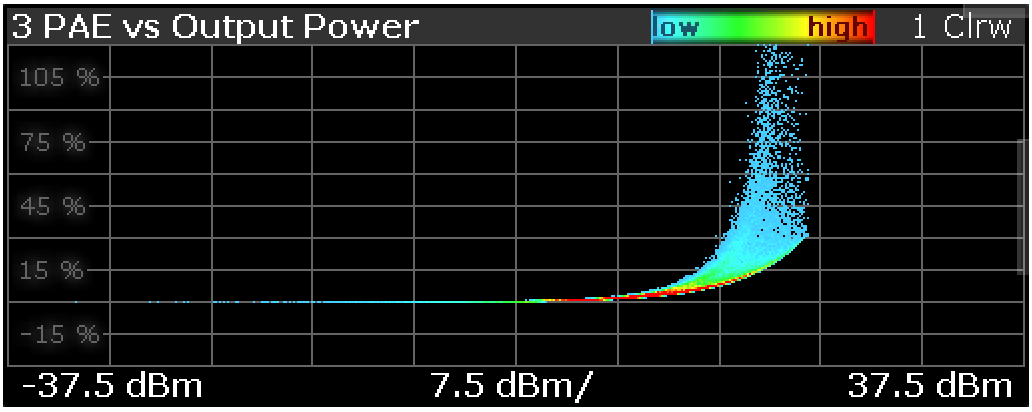
The "PAE vs Input Power" / "PAE vs Output Power" result displays show the power added efficiency (PAE) against the input or output power. It helps you to find the input or output levels at which the DUT works most efficiently.

The x-axis shows the levels of all samples of the reference signal (input power) or the measurement signal (output power) in dBm.

The y-axis shows the efficiency in %, based on the following formula:

PAE = (RF output power - RF input power) / DC power

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.



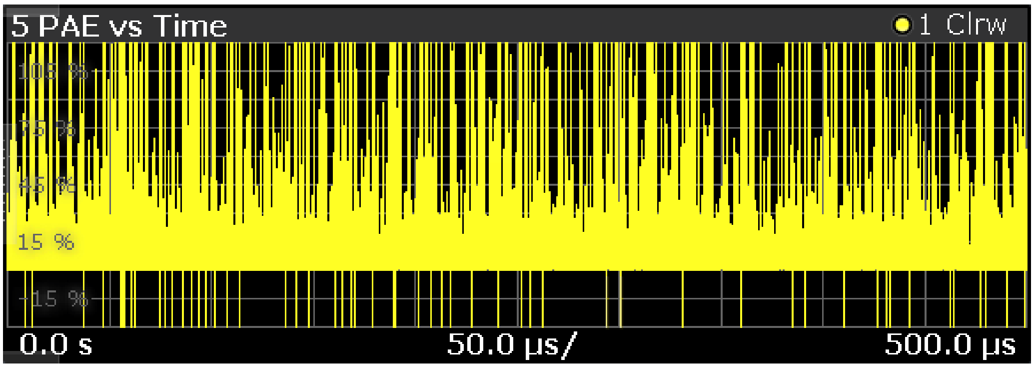
Remote command:
Selection: `LAY:ADD? '1',LEFT,PAEI`
Result query: `TRACe<n>[:DATA]?` on page 348

PAE vs Time
Note: This result display requires you to turn on the baseband input.

The "PAE vs Time" result display shows the power added efficiency against time.

The x-axis represents the time in seconds. The y-axis shows the efficiency in %, based on the following formula:

PAE = (RF output power - RF input power) / DC power



Remote command:
Selection: `LAY:ADD? '1',LEFT,PAET`
Result query: `TRACe<n>[:DATA]?` on page 348

Parameter Sweep Table

The "Parameter Sweep" result display shows a result parameter of the DUT (for example the EVM) against two (custom) measurement parameters.

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.

The parameter sweep table shows the minimum and maximum results for all available result types in numerical form. For each result type, the position at which the minimum and maximum result 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 % were measured (first and second row). The minimum EVM was measured at a frequency of 30 MHz and an output power of 0 dBm. The maximum EVM was measured at a frequency of 10 MHz and an output power of 0 dBm.

The following result types are evaluated in the parameter sweep.

- "ACLR Adj 1 Lower"
- "ACLR Adj 1 Upper"
- "ACLR Adj 1 Balanced"
- "ACLR Alt 1 Balanced"
- "ACLR Alt 2 Balanced"
- "AM/AM Curve Width"
- "AM/PM Curve Width"
- "Balanced ACLR Magnitude"
- Compression Point "P (1 dB / 2 dB / 3 dB)"
- "Crest Factor Out"
- "Current OBW"
- "EVM"
- "Gain"
- "Power Out"
- "RMS Power"
- "Voltage (V_cc)"
- "Current (I_cc)"
- "Power (V_cc * I_cc)"

For details on the parameters, see Chapter 2.2, "Amplifier parameters", on page 33.

For more information about how to select the parameters, see "Selecting the data to be evaluated during the parameter sweep" on page 133.

Remote command:
Selection: `LAY:ADD? '1',LEFT,PTAB`
Result query: Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367

Parameter Sweep: Diagram
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."

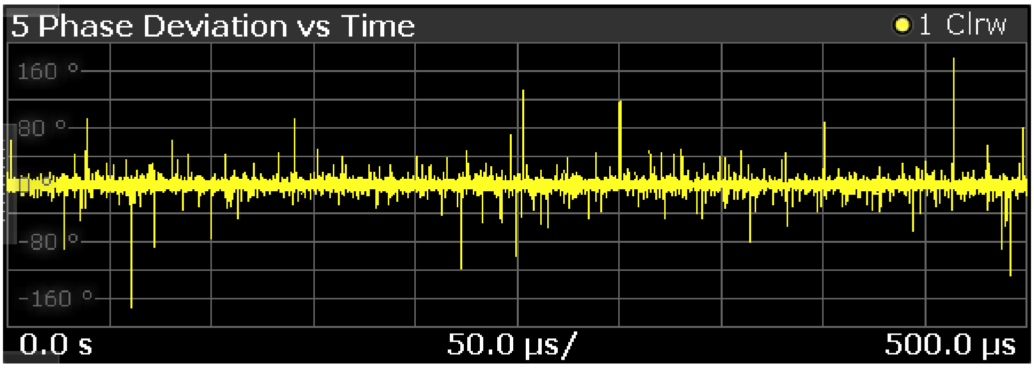
Remote command:
Selection: `LAY:ADD? '1',LEFT,PSW`
Result query: `TRACe<n>[:DATA]?` on page 348

Phase Deviation vs Time
The "Phase Deviation vs Time" result display shows the (unwrapped) 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 348

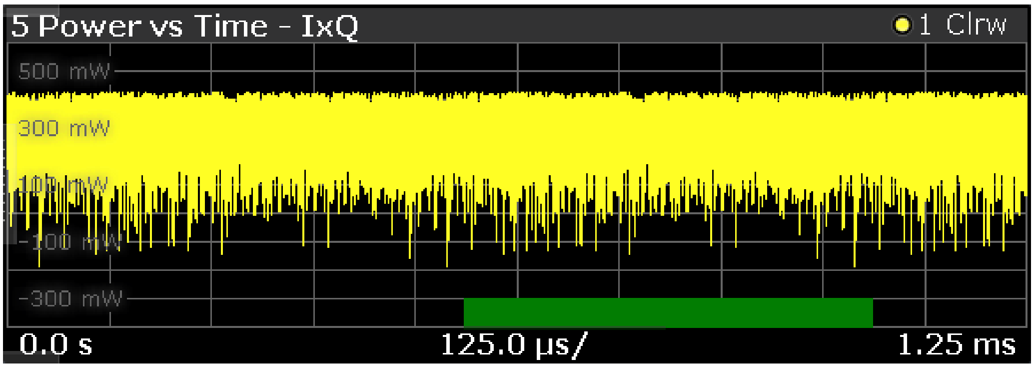
Power vs Time
Note: This result display requires you to turn on the baseband input.

The "Power vs Time" result display shows the supply power of the power amplifier against time.

The results are calculated by multiplying the supply voltage with the supply current which are recorded at the baseband inputs of the FSW.

The unit of the results is W.

For valid results, make sure that you have configured the measurement correctly regarding the equipment you are using.



Remote command:
Selection: LAY:ADD? '1',LEFT,PVT
Result query: TRACe<n>[:DATA]? on page 348

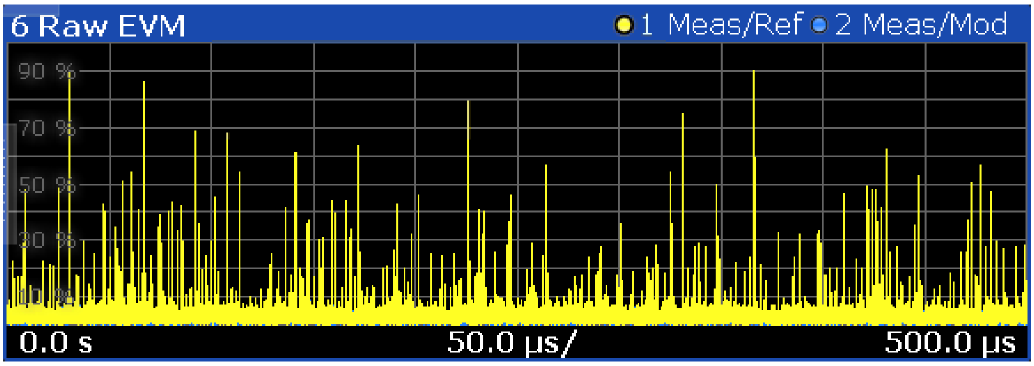
Raw EVM
The "Raw EVM" result display shows the error vector magnitude of the signal over time.

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 348

Numeric Result Summary

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

The table contains several areas.
- The modulation accuracy
- The power characteristics of the RF signal
- The power supply characteristics of the amplifier

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"
- All baseband results, except the "Average PAE"

Note: When synchronization fails or is disabled, some results can be unavailable.

For details on the parameters, see Chapter 2.2, "Amplifier parameters", on page 33.

Remote command:
Selecting the result display: `LAY:ADD? '1',LEFT,RTAB`
Querying results: see Chapter 5.8.2, "Retrieving numeric results", on page 350

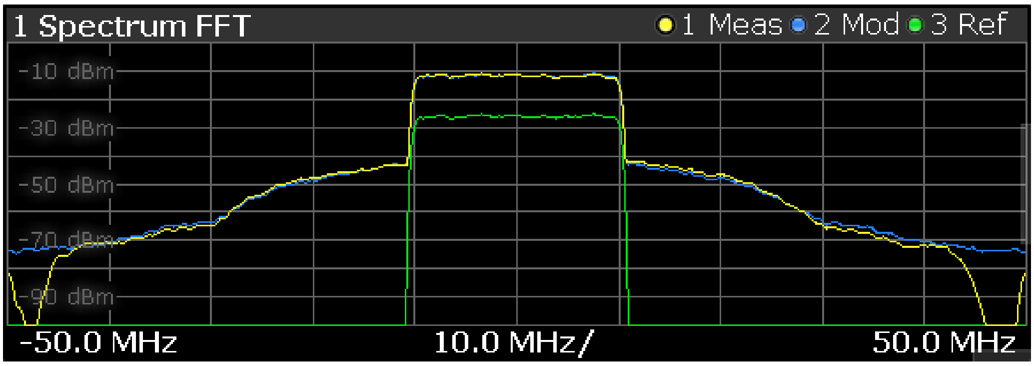
Spectrum FFT

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

It is available for the data recorded on the RF input and both baseband inputs (I and Q channels). (Note that the spectrum FFT of the I and Q channel are only available when parallel baseband capture has been turned on.)

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.



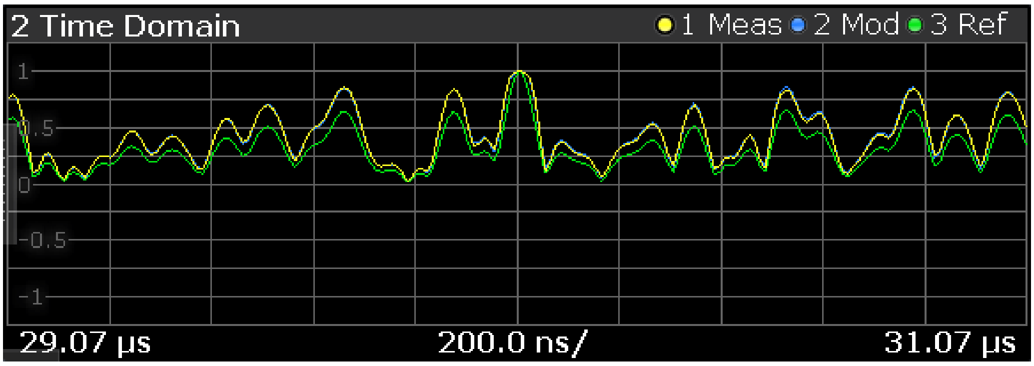
Remote command:
Selection (RF): `LAY:ADD? '1',LEFT,RFS`
Selection (I): `LAY:ADD? '1',LEFT,ISP`
Selection (Q): `LAY:ADD? '1',LEFT,QSP`
Result query: `TRACe<n>[:DATA]?` on page 348

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.
- Current measured at the I channel of the baseband input
  Trace 4 shows the characteristics of the current that is drawn by the amplifier. It is measured at the I channel of the baseband input.
- Voltage measured at the Q channel of the baseband input
  Trace 5 shows the characteristics of the power amplifier supply voltage. It is measured at the Q channel of the baseband input.
- Power measured at the baseband input
  Trace 6 shows the power of the signal at the baseband input. The power is the product of the current and the voltage measured at the baseband channels.
  Traces 4 to 6 are available when parallel baseband capture has been turned on.

For the baseband capture, all multipliers and offsets are already included in the results.



Remote command:
Selection: `LAY:ADD? '1',LEFT,TDOM`
Result query: `TRACe<n>[:DATA]?` on page 348

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 "Time domain result display" on page 152). 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 261. If statistics mode is switched off, the statistics table stays empty.

All statistics are usually evaluated with the corresponding linear power unit before the unit conversion is done and the final unit is displayed. To illustrate, in the case of EVM, this means that the mean EVM is equal to the square root of the mean error power. The same principle applies accordingly to all other values.

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 "Continuous Statistics" 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.

For details on the parameters, see Chapter 2.2, "Amplifier parameters", on page 33.

Remote command:
Adding statistics table: `LAY:ADD? '1',LEFT,STAB`
Querying results: Chapter 5.8.2.4, "Retrieving results of the statistics table", on page 374
Configuring statistics table: Chapter 5.7.4, "Configuring the statistics table", on page 335
Navigating through results ranges found in a capture: `CONFigure:RESult:RANGe[:SELected]` on page 261

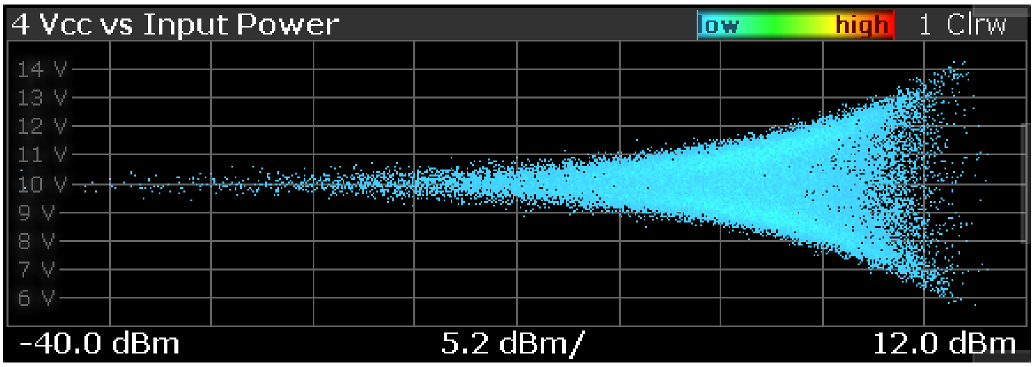
Vcc vs Power / Icc vs Power

Note: This result display requires you to turn on the baseband input.

The "Vcc vs Power" / "Icc vs Power" result displays show the voltage (Vcc) or current (Icc) against the input or output power. It helps you to find the input or output levels at which the DUT works most efficiently.

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 which signal characteristics you want to display in the result settings.

The y-axis shows the voltage or current, depending on the result display.



Remote command:
Selection: `LAY:ADD? '1',LEFT,PAEI`
Result query: `TRACe<n>[:DATA]?` on page 348

2.2 Amplifier parameters

The following parameters characterize the amplifier and are determined during an amplifier measurement.

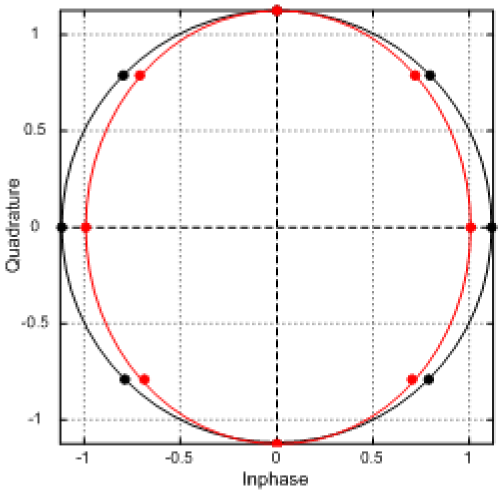
2.2.1 Modulation accuracy parameters

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 3.10, "Signal error estimation and compensation", on page 105.

Remote command:
`FETCh:MACCuracy:ADRoop:CURRent[:RESult]?` on page 353
`FETCh:STABle:ADRoop:STDeviation?` on page 380

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 3.10, "Signal error estimation and compensation", on page 105.

Remote command:
`FETCh:MACCuracy:GIMBalance:CURRent[:RESult]?` on page 353
`FETCh:STABle:GIMBalance:STDeviation?` on page 387

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 3.10, "Signal error estimation and compensation", on page 105.

Remote command:
`FETCh:MACCuracy:FERRor:CURRent[:RESult]?` on page 353
`FETCh:STABle:FERRor:STDeviation?` on page 386

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 3.10, "Signal error estimation and compensation", on page 105.

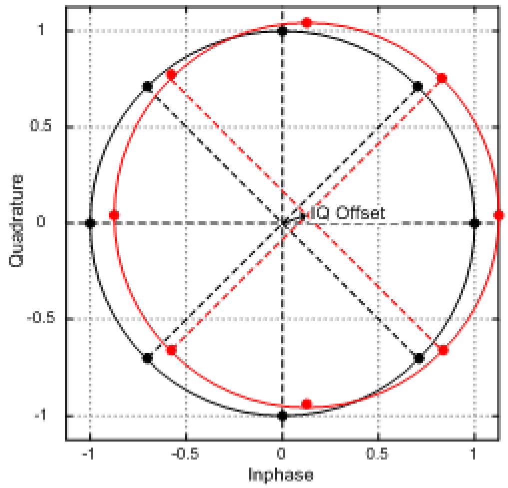
Remote command:

FETCh:MACCuracy:IQIMbalance:CURRent[:RESult]? on page 354

FETCh:STABle:IQIMbalance:STDeviation? on page 389

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 3.10, "Signal error estimation and compensation", on page 105.

Remote command:

FETCh:MACCuracy:IQOFfset:CURRent[:RESult]? on page 354

FETCh:STABle:IQOFfset:STDeviation? on page 389

Magnitude Error

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

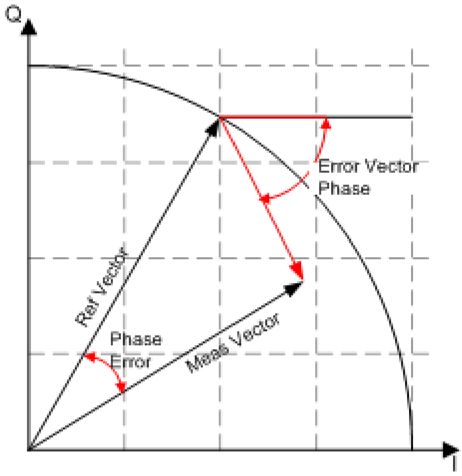
Remote command:

FETCh:MACCuracy:MERRor:CURRent[:RESult]? on page 354

FETCh:STABle:MERRor:STDeviation? on page 391

Phase Error

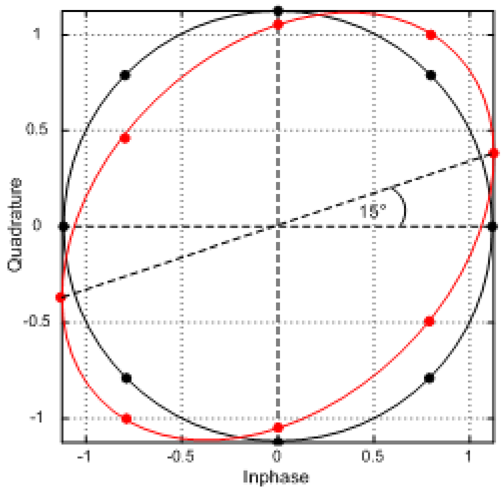
Phase difference between the reference and the (unwrapped) measured signal.



Remote command:
`FETCh:MACCuracy:PERRor:CURRent[:RESult]?` on page 355
`FETCh:STABle:PERRor:STDeviation?` on page 395

Quadrature Error

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



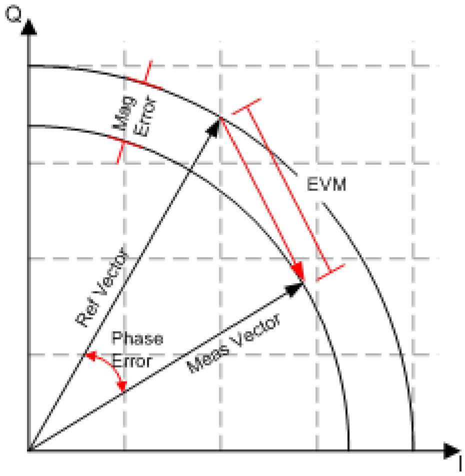
Within an ideal transmitter, the I and Q signal parts are mixed with an angle of 90° by the I/Q output mixer. Due to hardware imperfections, the signal delay of I and Q can be different and thus lead to an angle non-equal to 90°.

Note that quadrature rate error is not available if the I/Q Imbalance estimation is switched off. See also Chapter 3.10, "Signal error estimation and compensation", on page 105.

Remote command:
`FETCh:MACCuracy:QERRor:CURRent[:RESult]?` on page 356
`FETCh:STABle:QERRor:STDeviation?` on page 399

Raw EVM

Error vector magnitude between synchronized reference and measured signal.



Remote command:
FETCh:MACCuracy:REVM:CURRent[:RESult]? on page 356
FETCh:STABle:REVM:AVG:STDeviation? on page 401
FETCh:STABle:REVM:MAX:STDeviation? on page 401
FETCh:STABle:REVM:MIN:STDeviation? on page 402
FETCh:PTABle:EVM..., see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.

Raw Model EVM

Error vector magnitude between synchronized measured and model signal.

Remote command:
FETCh:MACCuracy:RMEV:CURRent[:RESult]? on page 356
FETCh:STABle:RMEV:AVG:STDeviation? on page 402
FETCh:STABle:RMEV:MAX:STDeviation? on page 403
FETCh:STABle:RMEV:MIN:STDeviation? on page 403

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 3.10, "Signal error estimation and compensation", on page 105.

Remote command:
FETCh:MACCuracy:SRERror:CURRent[:RESult]? on page 357
FETCh:STABle:SRERor:STDeviation? on page 404

2.2.2 Power characteristics

ACLR Tx
Power of the transmission channel.

Remote command:
`FETCh:PTABle:ACP...`, see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.

ACLR Adj Upper
Power of the adjacent upper channels.

Remote command:
`FETCh:PTABle:ACP:AChannel<ch>:UPP...`, see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.

ACLR Adj Lower
Power of the lower adjacent channels.

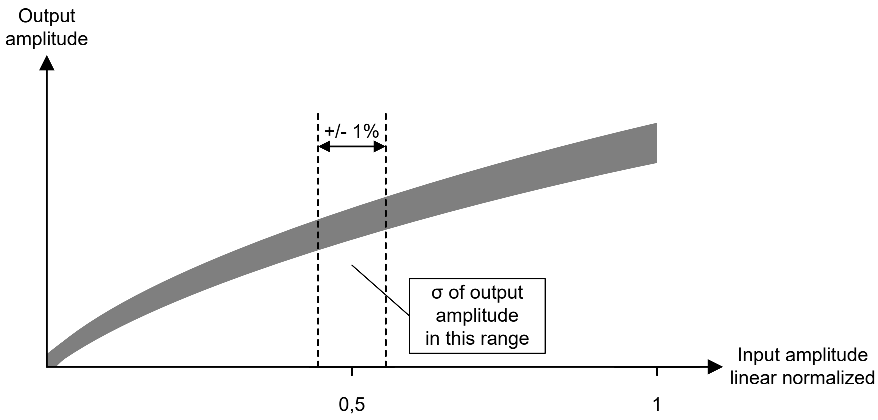
Remote command:
`FETCh:PTABle:ACP:AChannel<ch>:LOW...`, see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.

Balanced ACLR Magnitude
Difference between the lower and upper adjacent channel power.

Remote command:
`FETCh:PTABle:ACP:BALanced...`, see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.

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.



Remote command:
FETCh:AMAM:CWIDth:CURRent[:RESult]? on page 358
FETCh:STABle:AMAM:CWIDth:STDeviation? on page 381
FETCh:PTABle:AMAM:CWIDth..., see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.

AM/AM Curve Width (Pk-Pk)

The "AM/AM" display shows the peak curve width around the specified reference value.

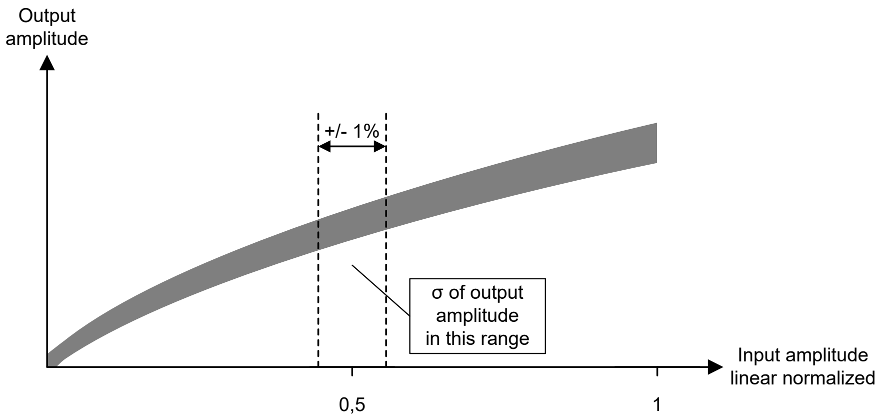
The values are Pk-Pk values, and represent max(output power/dBm) - min(output power/ dBm).

Remote command:
FETCh:AMAM:PEAK:CWIDth:CURRent[:RESult]? on page 360

AM/PM Curve Width

Vertical spread of the samples in the "Phase Deviation vs Input Power" ("AM/PM") result display.

The "Phase Deviation vs Input Power" 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.



Remote command:
`FETCh:AMPM:CWIDth:CURRent[:RESult]?` on page 359
`FETCh:STABle:AMPM:CWIDth:STDeviation?` on page 381
`FETCh:PTABle:AMPM:CWIDth...`, see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.

AM/PM Curve Width (Pk-Pk)
The "AM/PM Curve Width (Pk-Pk)" display shows the peak curve width around the specified reference value.

The values are Pk-Pk values, and represent max(phase deviation/°) - min(phase deviation/°).

Remote command:
`FETCh:AMPM:PEAK:CWIDth:CURRent[:RESult]?` on page 359

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 128).

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

Remote command:
`FETCh:POWer:P1DB:CURRent[:RESult]?` on page 363
`FETCh:STABle:P1DB:IN:STDeviation?` on page 392
`FETCh:PTABle:P1DB...`, see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.
`FETCh:POWer:P2DB:CURRent[:RESult]?` on page 363
`FETCh:STABle:P2DB:IN:STDeviation?` on page 393
`FETCh:PTABle:P2DB...`, see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.
`FETCh:POWer:P3DB:CURRent[:RESult]?` on page 363
`FETCh:STABle:P3DB:IN:STDeviation?` on page 394
`FETCh:PTABle:P3DB...`, see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.

Crest Factor In

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

Remote command:
`FETCh:POWer:CFACtor:IN:CURRent[:RESult]?` on page 360
`FETCh:STABle:CFACtor:IN:STDeviation?` on page 385

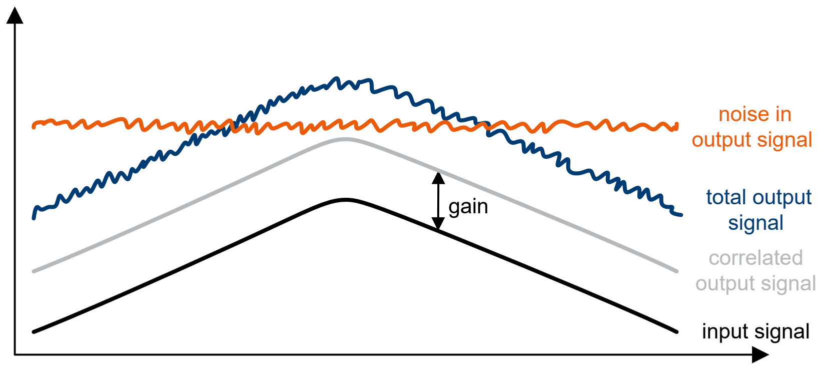
Crest Factor Out

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

Remote command:
`FETCh:POWer:CFACtor:OUT:CURRent[:RESult]?` on page 361
`FETCh:STABle:CFACtor:OUT:STDeviation?` on page 385
`FETCh:PTABle:CFACtor...`, see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.

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.

Remote command:
`FETCh:POWer:GAIN:CURRent[:RESult]?` on page 361
`FETCh:STABle:GAIN:STDeviation?` on page 386
`FETCh:PTABle:GAIN...`, see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.

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.

Remote command:
`FETCh:POWer:P1DB:OUT:CURRent[:RESult]?` on page 363
`FETCh:STABle:P1DB:OUT:STDeviation?` on page 392
`FETCh:POWer:P2DB:OUT:CURRent[:RESult]?` on page 363
`FETCh:STABle:P2DB:OUT:STDeviation?` on page 393
`FETCh:POWer:P3DB:OUT:CURRent[:RESult]?` on page 363
`FETCh:STABle:P3DB:OUT:STDeviation?` on page 394

Occupied Bandwidth
Occupied bandwidth calculated for the defined evaluation range.

Remote command:
`FETCh:POWer:OBW:CURRent[:RESult]?` on page 362

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.

Remote command:
`FETCh:POWer:INPut:CURRent[:RESult]?` on page 361
`FETCh:STABle:POWer:INPut:AVG:STDeviation?` on page 396
`FETCh:STABle:POWer:INPut:MAX:STDeviation?` on page 396
`FETCh:STABle:POWer:INPut:MIN:STDeviation?` on page 397

Power Out
Signal power at the DUT output.

It is the RMS power of:
- The currently selected frame, if FSW-K18 has successfully synchronized.
- The current capture buffer, if FSW-K18 has not synchronized.

Remote command:
`FETCh:POWer:OUTPut:CURRent[:RESult]?` on page 362
`FETCh:STABle:POWer:OUTPut:AVG:STDeviation?` on page 397
`FETCh:STABle:POWer:OUTPut:MAX:STDeviation?` on page 398
`FETCh:STABle:POWer:OUTPut:MIN:STDeviation?` on page 398
`FETCh:PTABle:POUT...`, see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.

Power Out (Sensor)
Signal power at the output power sensor.

Remote command:
`FETCh:POWer:SENSor:OUT:CURRent[:RESult]?` on page 364

RMS Power
RMS signal power at the DUT output.

Remote command:
FETCh:PTABle:RMS..., see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.

2.2.3 Power supply characteristics

These results are available when you enable baseband input.

For valid results, make sure that you have configured the measurement correctly regarding the equipment you are using (see "Configuring PAE measurements (Power Added Efficiency)" on page 126).

Baseband Input Voltage I
Voltage measured at the I channel of the analyzer baseband input.

Remote command:
FETCh:IVOLtage:PURE:CURRent[:RESult]? on page 366
FETCh:STABle:IVOLtage:AVG:STDeviation? on page 390
FETCh:STABle:IVOLtage:MAX:STDeviation? on page 390
FETCh:STABle:IVOLtage:MIN:STDeviation? on page 391

Baseband Input Voltage Q
Voltage measured at the Q channel of the analyzer baseband input.

Remote command:
FETCh:QVOLtage:PURE:CURRent[:RESult]? on page 366
FETCh:STABle:QVOLtage:AVG:STDeviation? on page 399
FETCh:STABle:QVOLtage:MAX:STDeviation? on page 400
FETCh:STABle:QVOLtage:MIN:STDeviation? on page 400

Voltage
Voltage measured at the Q channel of the analyzer baseband input.

This value represents the supply voltage of the power amplifier.

The result is the same as the "Baseband Input Voltage Q" when the multiplier = 1 and the offset = 0 (see "Configuring PAE measurements (Power Added Efficiency)" on page 126)

Remote command:
FETCh:VCC:CURRent[:RESult]? on page 367
FETCh:STABle:VCC:AVG:STDeviation? on page 404
FETCh:STABle:VCC:MAX:STDeviation? on page 405
FETCh:STABle:VCC:MIN:STDeviation? on page 405
FETCh:PTABle:VCC..., see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.

Current

Current measured at the I channel of the baseband input. This corresponds to the current drawn by the amplifier.

Remote command:
FETCh:ICC:CURRent[:RESult]? on page 366
FETCh:STABle:ICC:AVG:STDeviation? on page 387
FETCh:STABle:ICC:MAX:STDeviation? on page 388
FETCh:STABle:ICC:MIN:STDeviation? on page 388
FETCh:PTABle:ICC..., see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.

Power

DC power measured at the baseband input. The DC power is the product of the measured voltage and current.

Remote command:
FETCh:BBPower:CURRent[:RESult]? on page 365
FETCh:STABle:BBPower:AVG:STDeviation? on page 383
FETCh:STABle:BBPower:MAX:STDeviation? on page 384
FETCh:STABle:BBPower:MIN:STDeviation? on page 384

Average PAE

The average Power Added Efficiency (PAE) indicates the efficiency of the amplifier.

The PAE is the ratio of the difference between RF output and input power and the DC power:

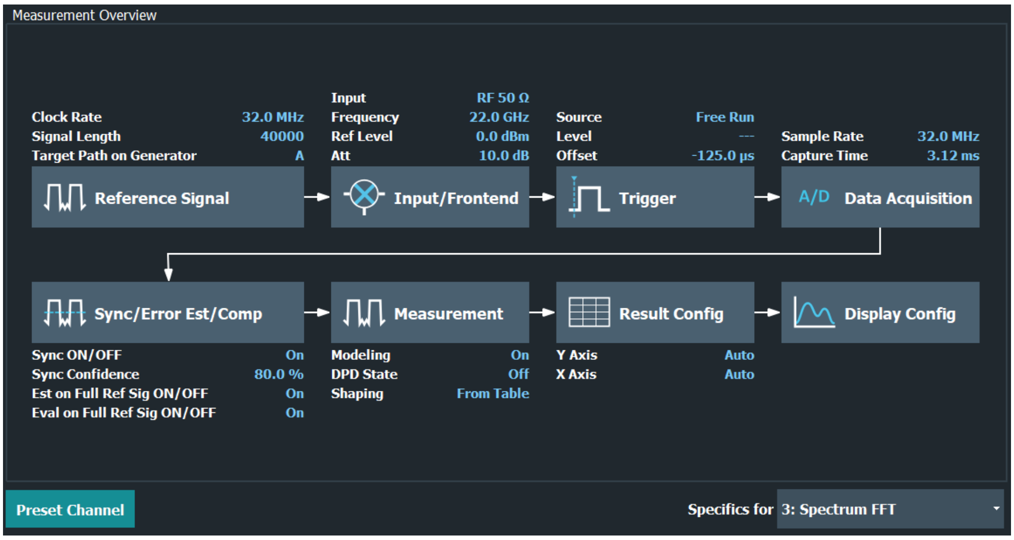
PAE = (Output Power - Input Power) / DC power

Remote command:
FETCh:APAE:CURRent[:RESult]? on page 365
FETCh:STABle:APAE:AVG:STDeviation? on page 382
FETCh:STABle:APAE:MAX:STDeviation? on page 382
FETCh:STABle:APAE:MIN:STDeviation? on page 383
FETCh:PTABle:PAE..., see Chapter 5.8.2.3, "Retrieving results of the parameter sweep table", on page 367.

3 Configuration

3.1 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 3.3, "Reference signals", on page 48.
2. Input and output
   See Chapter 3.4, "Inputs and outputs", on page 60.
3. Trigger
   See Chapter 3.5, "Triggering", on page 95.
4. Data Acquisition
   See Chapter 3.6, "Data acquisition", on page 95.
5. Synchronization, error estimation and compensation
   See Chapter 3.8, "Synchronization", on page 100.
   See Chapter 3.10, "Signal error estimation and compensation", on page 105.
6. Measurement
   Modeling: see Chapter 3.12, "System models", on page 108.
   DPD: see Chapter 3.13, "Digital predistortion", on page 111.
   Envelope: see Chapter 3.15, "Envelope measurements", on page 126.
7. Result configuration
   See Chapter 4, "Analysis", on page 140.
8. Display configuration
   See Chapter 2, "Measurements and result displays", on page 12.

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

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

Remote command:
`SYSTem:PRESet:CHANnel[:EXEC]` on page 168

Specific Settings for
The channel can 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.

3.2 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 5.5.1, "Performing measurements", on page 176.

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

While the measurement is running, "Continuous Sweep" and [RUN CONT] 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, "Continuous Sweep" 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, [RUN CONT] controls the Sequencer, not individual sweeps. [RUN CONT] starts the Sequencer in continuous mode.

For details on the Sequencer, see the R&S FSW user manual.

Remote command:
`INITiate<n>:CONTinuous` on page 177

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

Note: Sequencer. If the Sequencer is active, "Single Sweep" 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, the Sequencer sweeps a channel in single sweep mode only once.
Furthermore, [RUN SINGLE] controls the Sequencer, not individual sweeps. [RUN SINGLE] starts the Sequencer in single mode.

If the Sequencer is off, only the evaluation for the currently displayed channel is updated.

For details on the Sequencer, see the R&S FSW user manual.

Remote command:
`INITiate<n>[:IMMediate]` on page 177

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

Remote command:
`INITiate<n>:CONMeas` on page 177

3.3 Reference signals

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
  (Requires a Rohde & Schwarz generator.)
- Designing the signal in a waveform file
- Designing the signal in the amplifier application
  (Requires a Rohde & Schwarz generator.)
- Designing the signal with a Crest Factor Reduction (Generator Option K548)

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

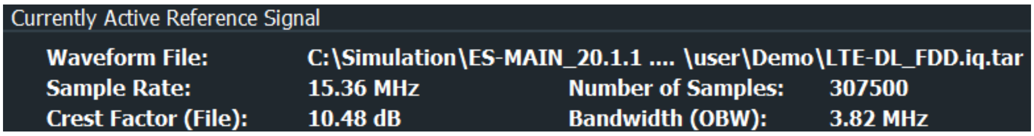
The remote commands required to configure the reference signal are described in Chapter 5.6.1, "Designing a reference signal", on page 180.

3.3.1 Reference signal information

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 188
Sample rate: `CONFigure:REFSignal:SINFo:SRATe?` on page 189
Sample length: `CONFigure:REFSignal:SINFo:SLENgth?` on page 189
Crest Factor: `CONFigure:REFSignal:SINFo:CFACtor?` on page 189
OBW: `CONFigure:REFSignal:SINFo:OBW?` on page 190
reference trace I/Q data: `TRACe:IQ:REF[:DATA]?` on page 406

3.3.2 Using multi-segment waveform files

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 188

3.3.3 Transferring the reference signal

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 3.4.7, "Controlling a signal generator", on page 79.

- 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 FSW-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 FSW-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.
  Check if the generator IP address / computer name are correct and if the connection has been established.

3.3.4 Current Generator Waveform

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

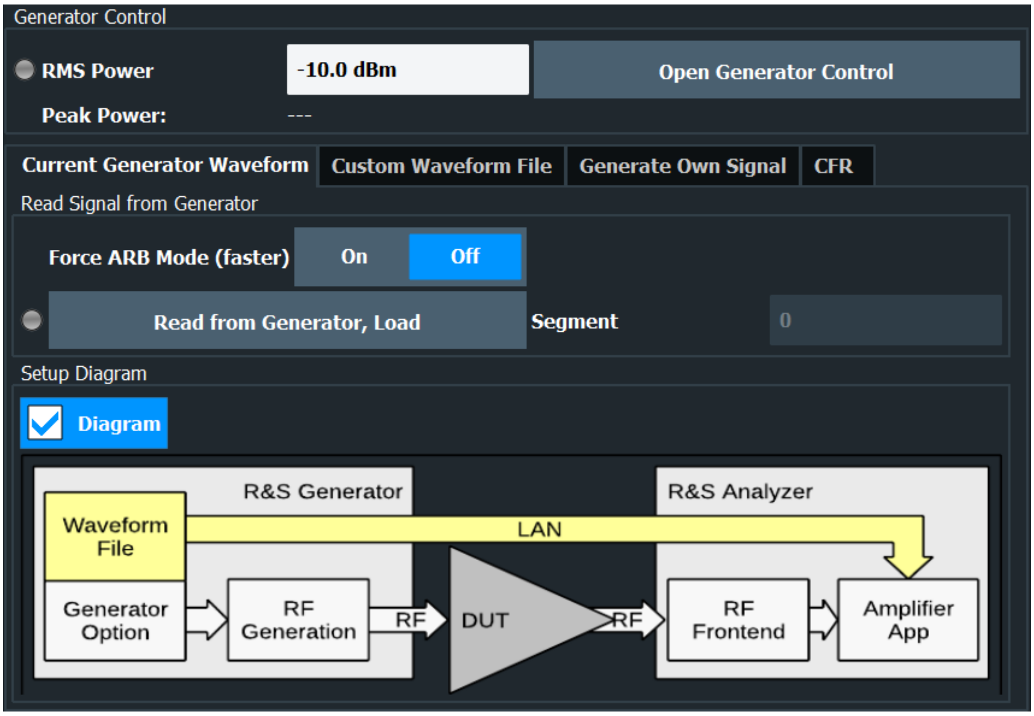
Designing a reference signal on a signal generator
One way to design a reference signal is to design the signal on the signal generator itself.

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. Also the waveform header of the file is recalculated to make sure that the peak input level stays constant or lower than the gain expansion during DPD calculation.

The parameters of the currently active reference signal are described in "Reference signal information" on page 49.

The "Open Generator Control" button provides functionality to change the generator settings as described in Chapter 3.4.7, "Controlling a signal generator", on page 79.



Most of the options available for the connected generator are supported by the automatic signal import functionality of the R&S FSW-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 181
`CONFigure:REFSignal:CGW:READ` on page 182
`CONFigure:REFSignal:CGW:LEDState?` on page 181

3.3.5 Custom Waveform File

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

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.

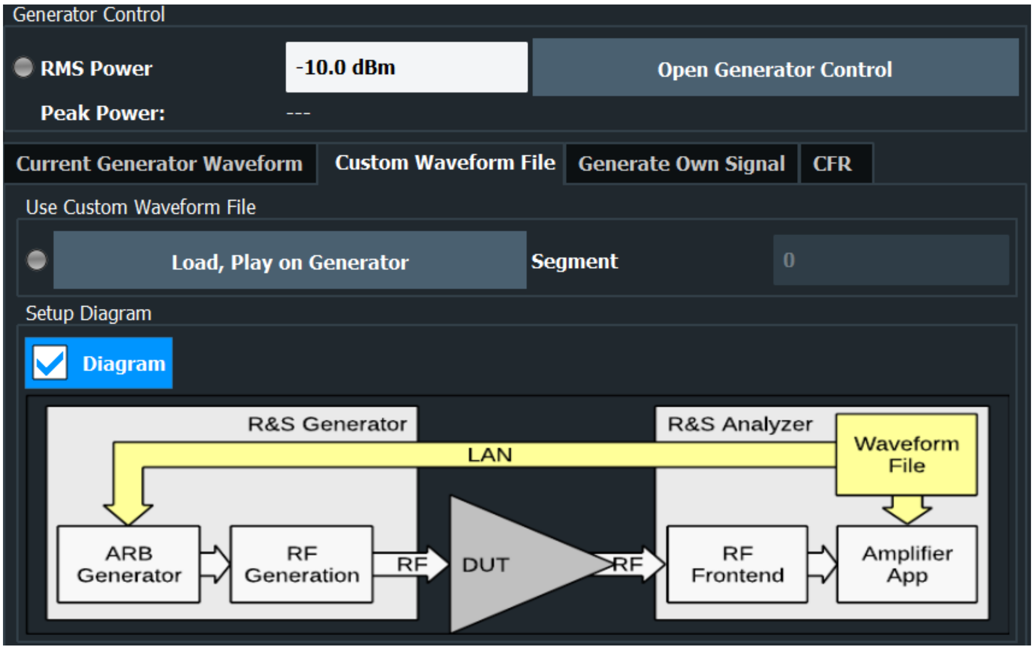
There are two ways to generate the reference signal through a custom waveform file.

- The generator is connected to the FSW in a LAN, and can be recognized by the R&S FSW-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 FSW-K18. The generator then generates the corresponding signal with the appropriate signal level, and the R&S FSW-K18 is able to compare the measured signal to the ideal reference signal.
- The generator is not connected to the FSW
  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 49.

The "Open Generator Control" button provides functionality to change the generator settings as described in Chapter 3.4.7, "Controlling a signal generator", on page 79.

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 "Load, Play on Generator".

► 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 183
Transfer file: CONFigure:REFSignal:CWF:WRITe on page 184
Transmission state: CONFigure:REFSignal:CWF:LEDState? on page 183
Export file: CONFigure:REFSignal:CWF:ETGenerator[:STATe] on page 182
DUT input power: CONFigure:REFSignal:CWF:DPIPower on page 182

3.3.6 Generate Own Signal

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

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

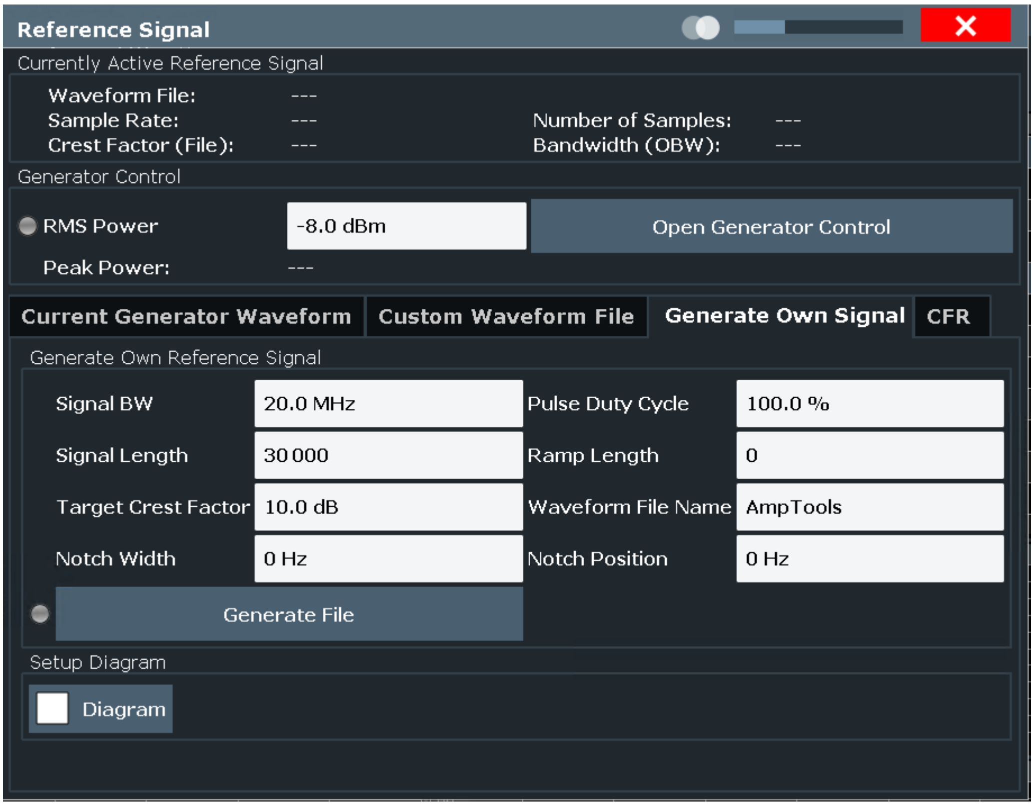
The generated signal is a pseudo-noise signal, whose basic properties, like crest factor and bandwidth, you can specify as required.

The parameters of the currently active reference signal are described in "Reference signal information" on page 49.

The "Open Generator Control" button provides functionality to change the generator settings as described in Chapter 3.4.7, "Controlling a signal generator", on page 79.

Table 3-1: Parameter dependencies

Parameter	Min Value	Max Value
Target Crest Factor	2	13
Signal Length	((N+2*RampLength)*100)/PulseDutyCycle N=1000 for Target Crest Factor < 12.5 dB N=25000 for Target Crest Factor ≥ 12.5 dB	
Notch Width	Signal Bandwidth/100	Signal Bandwidth
Ramp Length		0 if Pulse Duty Cycle is 100 % Otherwise (Signal Length*PulseDutyCycle/100-N)/2



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 55
- "Pulse Duty Cycle" on page 56
- "Signal Length" on page 56
- "Ramp Length" on page 56
- "Target Crest Factor" on page 56
- "Waveform File Name" on page 56
- "Notch Width" on page 56
- "Notch Position" on page 57

Remote command:

`CONFigure:REFSignal:GOS:WRITe` on page 188
`CONFigure:REFSignal:GOS:LEDState?` on page 185

Signal Bandwidth ← Designing a reference signal within the R&S FSW-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 184

Pulse Duty Cycle ← Designing a reference signal within the R&S FSW-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 185

Signal Length ← Designing a reference signal within the R&S FSW-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 56.

Remote command:
`CONFigure:REFSignal:GOS:SLENgth` on page 187

Ramp Length ← Designing a reference signal within the R&S FSW-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 187

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

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

The crest factor is defined for a signal with 100 % pulse duty cycle and 0 Hz notch. Changes to the pulse duty cycle and notch parameters will change the crest factor.

Remote command:
`CONFigure:REFSignal:GOS:CRESt` on page 185

Waveform File Name ← Designing a reference signal within the R&S FSW-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 187

Notch Width ← Designing a reference signal within the R&S FSW-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 186

Notch Position ← Designing a reference signal within the R&S FSW-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 186

3.3.7 CFR (Crest Factor Reduction)

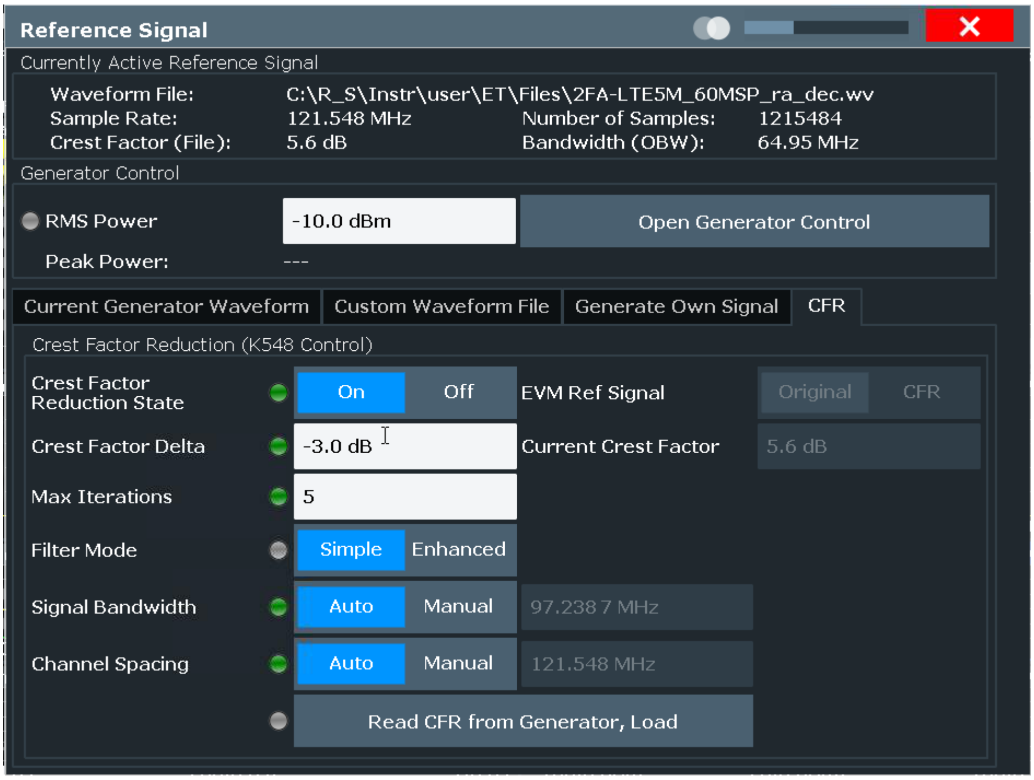
Access (source: K548 control): "Overview" > "Reference Signal" > "CFR"

Crest Factor Reduction (Generator Option K548)
The Crest Factor Reduction dialog provides functionality to control the main parameters of a Rohde & Schwarz signal generator equipped with option K548.

- CFR decreases the peak-to-average power ratio (PAPR) of the reference file, consequently leading to an increase in the EVM when compared to the reference signal.
- CFR often yields significantly improved DPD results due to reduced compression of the amplifier caused by less intense driving.
- As a result, the CFR provides a way to trade EVM for ACLR, balancing the two factors.

Note: During DPD, the CRF file needs to be selected as the reference.

For more detailed information, refer to the "Advanced Digital Pre-Distortion" video, which is part of the R&S amplifier masterclass playlist on the R&S youtube channel (Chapter 1, "Welcome to the amplifier measurement application", on page 9).



Crest Factor Reduction State ← Crest Factor Reduction (Generator Option K548)
Enables the crest factor reduction calculation.

If "On" is selected and all input fields are editable, crest factor reduction is active on the generator but the FSW-K18 application has no CFR reference yet.

If "On" is selected and "Read CFR from Generator, Load" is selected, all input fields are grayed out. The CFR reference can now be used by the FSW-K18 application.

If the state is changed to "Off", a potentially active DPD is also turned off. The DPD results will be invalid and new measurements will be needed.

Remote command:
`CONFigure:CFReduction[:STATe]` on page 190
`CONFigure:CFReduction[:STATe]:LEDState?` on page 190

EVM Ref. Signal ← Crest Factor Reduction (Generator Option K548)
Loads a new ARB file as reference if "CFR" is selected. The original ARB file is stored.

Remote command:
`CONFigure:CFReduction:RSORignal` on page 191

Crest Factor Delta ← Crest Factor Reduction (Generator Option K548)
Sets the value difference by which you want to change your crest factor.

Remote command:
`CONFigure:CFReduction:CFDelta` on page 193
`CONFigure:CFReduction:CFDelta:LEDState?` on page 193

Current Crest Factor ← Crest Factor Reduction (Generator Option K548)
Displays the current crest factor of the waveform after the calculation of the resulting crest factor is completed.

Remote command:
`CONFigure:CFReduction:CCFactor?` on page 193

Max Iterations ← Crest Factor Reduction (Generator Option K548)
Sets the number of iterations that are used for calculating the resulting crest factor. The iteration process is stopped when the desired crest factor delta is achieved by 0.1 dB.

Remote command:
`CONFigure:CFReduction:ITERations` on page 191
`CONFigure:CFReduction:ITERations:LEDState?` on page 191

Filter Mode ← Crest Factor Reduction (Generator Option K548)
Selects which filter mode is used for the filtering. In "Simple" mode, you can specify the RF bandwidth and channel spacing of the signal. The lowpass filter is designed to pass through frequency components inside the signal bandwidth and suppress components in the adjacent channel. In "Enhanced" mode, you can specify the passband and stop-band frequencies of the lowpass filter.

Remote command:
`CONFigure:CFReduction:FILTer` on page 191
`CONFigure:CFReduction:FILTer:LEDState?` on page 192

Signal Bandwidth ← Crest Factor Reduction (Generator Option K548)
Sets the signal bandwidth. The value of the "Signal Bandwidth" should not be higher than the "Channel Spacing".

When in automatic mode, "Signal Bandwidth" shall be set to the OBW value of the reference file (shown in reference signal overview).

"Channel Spacing" shall be set to 1.15 times this value.

Remote command:
`CONFigure:CFReduction:SBANdwidth` on page 190
`CONFigure:CFReduction:SBANdwidth:AUTO` on page 190
`CONFigure:CFReduction:SBANdwidth:LEDState?` on page 191

Channel Spacing ← Crest Factor Reduction (Generator Option K548)
Sets the channel spacing.

Remote command:
`CONFigure:CFReduction:CSPacing` on page 192
`CONFigure:CFReduction:CSPacing:AUTO` on page 192
`CONFigure:CFReduction:CSPacing:LEDState?` on page 192

Read CFR from Generator, Load ← Crest Factor Reduction (Generator Option K548)
Applies crest factor reduction on the connected signal generator.

Remote command:
`CONFigure:CFReduction:READ` on page 194
`CONFigure:CFReduction:READ:LEDState?` on page 194

Passband Frequency ← Crest Factor Reduction (Generator Option K548)
Sets the passband frequency. Only available for "Enhanced" filter mode.

Remote command:
`CONFigure:CFReduction:PFRequency` on page 195
`CONFigure:CFReduction:PFRequency:LEDState?` on page 195

Stopband Frequency ← Crest Factor Reduction (Generator Option K548)
Sets the stopband frequency. Only available for "Enhanced" filter mode.

Remote command:
`CONFigure:CFReduction:SFRequency` on page 195
`CONFigure:CFReduction:SFRequency:LEDState?` on page 195

Maximum Filter Order ← Crest Factor Reduction (Generator Option K548)
Sets the maximum filter order. Only available for "Enhanced" filter mode.

Remote command:
`CONFigure:CFReduction:MFORder` on page 194
`CONFigure:CFReduction:MFORder:LEDState?` on page 194

3.4 Inputs and outputs

3.4.1 Selecting and configuring the input source

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

The R&S FSW-K18 supports the RF input, the optional analog baseband input and optional input from external mixers.

Simultaneous use of the RF input and the Analog Baseband input

Compared to other applications available for the FSW, the FSW-K18 allows you to use both the RF input and the optional analog baseband input simultaneously.

This allows for various specific measurements which require a simultaneous capture of the RF signal, of the supply voltage and of the current drawn by an amplifier. Such a test setup is, for example, required to calculate the instantaneous PAE (Power Added Efficiency), which in turn is of interest for measurements on amplifiers that use envelope tracking.

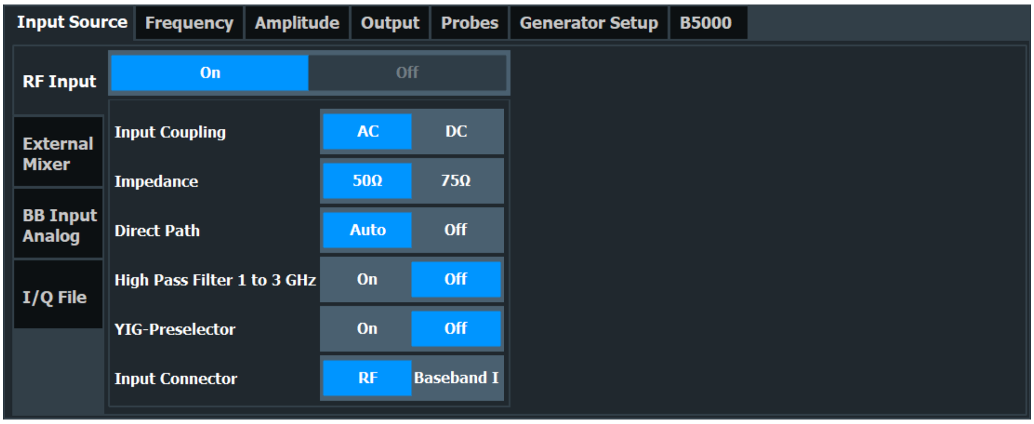
For more information about input from external mixers, refer to the FSW user manual.

3.4.1.1 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 FSW user manual.



The remote commands required to configure the RF input are described in Chapter 5.6.2, "Selecting and configuring the input source", on page 196.

Input Coupling

The RF input of the FSW can be coupled by alternating current (AC) or direct current (DC).

For an active external frontend, input coupling is always AC.

Not available for input from the optional "Analog Baseband" interface.

Not available for input from the optional "Digital Baseband" interface.

AC coupling blocks any DC voltage from the input signal. AC coupling is activated by default to prevent damage to the instrument. Very low frequencies in the input signal can 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 specifications document.

Remote command:
`INPut:COUPling` on page 197

Impedance

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

For an active external frontend, impedance is always 50 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).

Not available for input from the optional "Digital Baseband" interface.

Not available for input from the optional "Analog Baseband" interface. For analog baseband input, an impedance of 50 Ω is always used.

Remote command:
INPut:IMPedance on page 199

Direct Path

Enables or disables the use of the direct path for small frequencies.

In spectrum analyzers, passive analog mixers are used for the first conversion of the input signal. In such mixers, the LO signal is coupled into the IF path due to its limited isolation. The coupled LO signal becomes visible at the RF frequency 0 Hz. This effect is referred to as LO feedthrough.

To avoid the LO feedthrough the spectrum analyzer provides an alternative signal path to the A/D converter, referred to as the *direct path*. By default, the direct path is selected automatically for RF frequencies close to zero. However, this behavior can be disabled. If "Direct Path" is set to "Off", the spectrum analyzer always uses the analog mixer path.

For an active external frontend, the direct path is always used automatically for frequencies close to zero.

"Auto"	(Default) The direct path is used automatically for frequencies close to zero.
"Off"	The analog mixer path is always used.

Remote command:
INPut:DPATh on page 197

High Pass Filter 1 to 3 GHz

Activates an additional internal highpass filter for RF input signals from 1 GHz to 3 GHz. This filter is used to remove the harmonics of the analyzer to measure the harmonics for a DUT, for example.

This function requires an additional hardware option.

Note: For RF input signals outside the specified range, the high-pass filter has no effect. For signals with a frequency of approximately 4 GHz upwards, the harmonics are suppressed sufficiently by the YIG-preselector, if available.)

Remote command:
INPut:FILTer:HPASs[:STATe] on page 198

YIG-Preselector

Enables or disables the YIG-preselector.

This setting requires an additional option on the FSW.

An internal YIG-preselector at the input of the FSW ensures that image frequencies are rejected. However, image rejection 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 FSW, which can lead to image-frequency display.

Remote command:
INPut:FILTer:YIG[:STATe] on page 199

Input Connector
Determines which connector the input data for the measurement is taken from.

For more information on the optional "Analog Baseband" interface, see the FSW I/Q Analyzer and I/Q Input user manual.

"RF"	(Default:) The "RF Input" connector
"RF Probe"	The "RF Input" connector with an adapter for a modular probe This setting is only available if a probe is connected to the "RF Input" connector. It is not available for an active external frontend.
"Baseband Input I"	The optional "Baseband Input I" connector This setting is only available if the optional "Analog Baseband" interface is installed and active for input. It is not available for the FSW67. For FSW85 models with two input connectors, this setting is only available for "Input 1". This feature is available when you turn off Enable Parallel BB Capture.

Remote command:
`INPut:CONNector` on page 196

3.4.1.2 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.

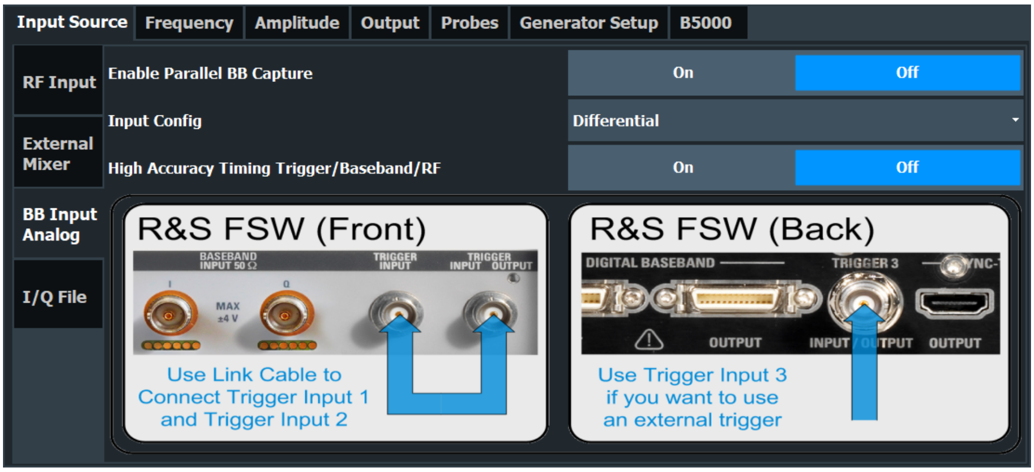
For more information about using external generators, refer to the FSW user manual.

3.4.1.3 Configuring the analog baseband input

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

The analog baseband input is available as a hardware option.

Measurements that also consider the supply voltage and the current drawn by the PA, require the analog baseband inputs to measure the voltage (baseband input Q) and the current (baseband input I). Typically, you need to connect some power probes to the baseband inputs for this purpose.



The remote commands required to configure the RF input are described in Chapter 5.6.2, "Selecting and configuring the input source", on page 196.

Enable Parallel BB Capture

Turns simultaneous data capture on the RF input and the analog baseband input on and off.

Simultaneous data capture is necessary when you measure the supply voltage and the current drawn by the PA.

Remote command:
`INPut<ip>:SELect:BBANalog[:STATe]` on page 200

Input Configuration

Defines whether the input is provided as a differential signal via all four Analog Baseband connectors or as a plain I/Q signal via two single-ended lines.

Note: Both single-ended and differential probes are supported as input; however, since only one connector is occupied by a probe, the "Single-ended" setting must be used for all probes.

"Single-ended"	I, Q data only
"Differential"	I, Q and inverse I,Q data (Not available for FSW85)

Remote command:
`INPut:IQ:BALanced[:STATe]` on page 199

High Accuracy Timing Trigger - Baseband - RF

Activates a mode with enhanced timing accuracy between analog baseband, RF and external trigger signals.

Note: Prerequisites for previous models of FSW.

For FSW models with a serial number lower than 103000, special prerequisites and restrictions apply for high accuracy timing:

- To obtain this high timing precision, trigger port 1 and port 2 must be connected via the Cable for High Accuracy Timing (order number 1325.3777.00).
- As trigger port 1 and port 2 are connected via the cable, only trigger port 3 can be used to trigger a measurement.
- Trigger port 2 is configured as output if the high accuracy timing option is active. Make sure not to activate this option if you use trigger port 2 in your measurement setup.
- When you first enable this setting, you are prompted to connect the cable for high accuracy timing to trigger ports 1 and 2. If you cancel this prompt, the setting remains disabled. As soon as you confirm this prompt, the cable must be in place - the firmware does not check the connection. (In remote operation, the setting is activated without a prompt.)

For more information, see the FSW I/Q Analyzer and I/Q Input User Manual.

Remote command:
`CALibration:AIQ:HATiming[:STATe]` on page 196

3.4.1.4 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 (FSW-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:SELect on page 199

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 FSW 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:FILE:PATH on page 197

File Repetitions

Determines how often the data stream is repeatedly copied in the I/Q data memory to create a longer record. If the available memory is not sufficient for the specified number of repetitions, the largest possible number of complete data streams is used.

Remote command:
TRACe:IQ:FILE:REPetition:COUNt on page 202

Selected Channel

Only available for files that contain more than one data stream from multiple channels: selects the data stream to be used as input for the currently selected channel.

In "Auto" mode (default), the first data stream in the file is used as input for the channel. Applications that support multiple data streams use the first data stream in the file for the first input stream, the second for the second stream etc.

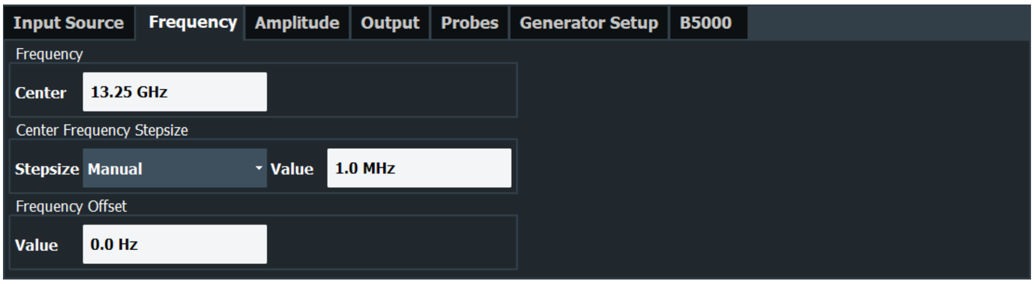
Remote command:
MMEMory:LOAD:IQ:STReam on page 201
MMEMory:LOAD:IQ:STReam:AUTO on page 201
MMEMory:LOAD:IQ:STReam:LIST? on page 202

3.4.2 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 FSW user manual.



The remote commands required to configure the frequency are described in Chapter 5.6.4, "Configuring the frequency", on page 236.

Center Frequency
Defines the frequency of the measured signal.

The possible value range depends on the FSW model you have. See the specifications document for more information about the supported frequency range.

Remote command:
`[SENSe:]FREQuency:CENTer` on page 236

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 237

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, 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.

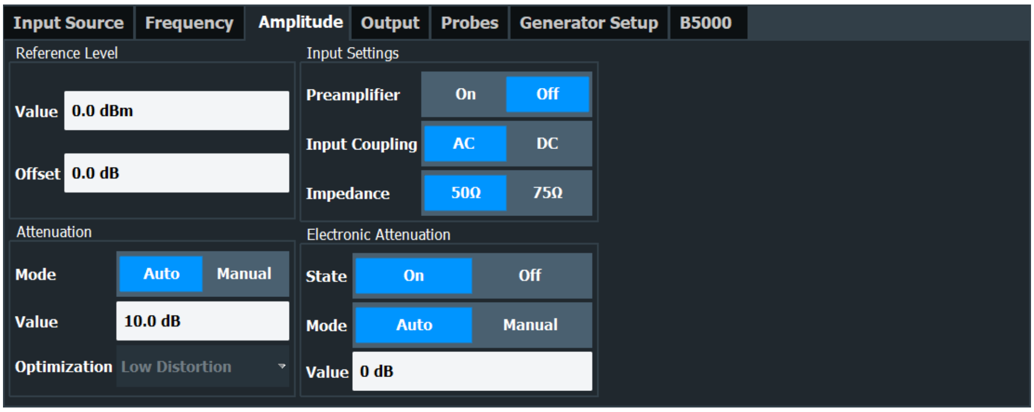
Remote command:
`[SENSe:]FREQuency:OFFSet` on page 237

3.4.3 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 FSW user manual.



The remote commands required to configure the amplitude are described in Chapter 5.6.5, "Defining level characteristics", on page 237.

Functions available in the "Amplitude" dialog box described elsewhere:
- "Input Coupling" on page 62
- "Impedance" on page 62

Reference Level

Defines the expected maximum reference level. Signal levels above this value are possibly not measured correctly. Signals above the reference level are indicated by an "IF Overload" or "OVLD" status display.

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

Since the hardware of the FSW 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 optimal measurement (no compression, good signal-to-noise ratio).

For an active external frontend, the reference level refers to the RF input at the external frontend, not the levels at the RF input of the FSW. The hardware is adjusted to the defined reference level optimally for input signals with a crest factor of 10 dB. Thus, the required reference level for an optimal measurement can differ depending on the crest factor of the input signal.

Remote command:
`DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:RLEVel`
on page 238

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 FSW 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 FSW-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 FSW must handle. Do not rely on the displayed reference level (internal reference level = displayed reference level - offset).

Remote command:
`DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:RLEVel:OFFSet` on page 238

Full Scale Level

The full scale level defines the maximum power you can input at the Baseband Input connector without clipping the signal.

- 0.25 V
- 0.5 V
- 1 V
- 2 V

If probes are connected, the possible full scale values are adapted according to the probe's attenuation and maximum allowed power.

Available for parallel capture on the baseband and RF inputs.

Remote command:
`INPut:IQ:FULLscale[:LEVel]` on page 244

Preamplifier

If the (optional) internal preamplifier hardware is installed on the FSW, a preamplifier can be activated for the RF input signal.

You can use a preamplifier to analyze signals from DUTs with low output power.

Note: If an optional external preamplifier is activated, the internal preamplifier is automatically disabled, and vice versa.

For an active external frontend, a preamplifier is not available.

This function is not available for input from the (optional) "Digital Baseband" interface.

For all FSW models except for FSW85, the following settings are available:

"Off"	Deactivates the preamplifier.
"15 dB"	The RF input signal is amplified by about 15 dB.
"30 dB"	The RF input signal is amplified by about 30 dB.

For older FSW43/FSW50/FSW67 models, the input signal is always ampilfied by about 30 dB when the preamplifier is active.

For FSW85 models, no preamplifier is available.

Remote command:
INPut:GAIN:STATe on page 242
INPut:GAIN[:VALue] on page 243

Ext. PA Correction

This function is only available if an external preamplifier is connected to the FSW, and only for frequencies above 1 GHz. For details on connection, see the preamplifier's documentation.

Using an external preamplifier, you can measure signals from devices under test with low output power, using measurement devices which feature a low sensitivity and do not have a built-in RF preamplifier.

When you connect the external preamplifier, the FSW reads out the touchdown (`.S2P`) file from the EEPROM of the preamplifier. This file contains the s-parameters of the preamplifier. As soon as you connect the preamplifier to the FSW, the preamplifier is permanently on and ready to use. However, you must enable data correction based on the stored data explicitly on the FSW using this setting.

When enabled, the FSW automatically compensates the magnitude and phase characteristics of the external preamplifier in the measurement results. Any internal preamplifier, if available, is disabled.

For FSW85 models with two RF inputs, you can enable correction from the external preamplifier for each input individually, but not for both at the same time.

When disabled, no compensation is performed even if an external preamplifier remains connected.

Remote command:
INPut:EGAin[:STATe] on page 242

Input Coupling

The RF input of the FSW can be coupled by alternating current (AC) or direct current (DC).

For an active external frontend, input coupling is always AC.

Not available for input from the optional "Analog Baseband" interface.

Not available for input from the optional "Digital Baseband" interface.

AC coupling blocks any DC voltage from the input signal. AC coupling is activated by default to prevent damage to the instrument. Very low frequencies in the input signal can 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 specifications document.

Remote command:
INPut:COUPling on page 197

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

For an active external frontend, impedance is always 50 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).

Not available for input from the optional "Digital Baseband" interface.

Not available for input from the optional "Analog Baseband" interface. For analog baseband input, an impedance of 50 Ω is always used.

Remote command:
INPut:IMPedance on page 199

Attenuation Mode / Value
Defines the attenuation applied to the RF input of the FSW.

This function is not available for input from the optional "Digital Baseband" interface.

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

Note: Using an external frontend. If an external frontend is active, you can configure the attenuation for the external frontend and the analyzer separately. When using an external frontend, only mechanical attenuation is available.
For more information, see the R&S FSW user manual.

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 specifications document. 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 can lead to hardware damage.

Remote command:
INPut:ATTenuation on page 239
INPut:ATTenuation:AUTO on page 239
Defining attenuation for the analyzer when using an external frontend:
INPut:SANalyzer:ATTenuation:AUTO on page 240
INPut:SANalyzer:ATTenuation on page 240

Optimization
Selects the priority for signal processing *after* the RF attenuation has been applied.

This function is only available under the following conditions:

- One of the following options that provide a separate wideband processing path in the FSW is installed:
  – Bandwidth extension R&S FSW-B160/-B320 Extension Board 1, Revision 2 or higher
  – Bandwidth extension R&S FSW-B512, B1200, B2001, B4001, B6001, or B8001
  – Real-time option R&S FSW-B160R
    (Currently not supported for K161R, B512R and B800R/K800RE)
- An I/Q bandwidth that requires the wideband path is used.
- The optional "Digital Baseband" interface is not active.

"Low distortion"
(Default:) Optimized for low distortion by avoiding intermodulation

"Low noise" Optimized for high sensitivity and low noise levels
If this setting is selected, "Low noise" is indicated in the channel information bar.

Remote command:
INPut:ATTenuation:AUTO:MODE on page 241

Signal Path
Selects the signal path for signal processing.

"Narrowband" (Default:) The narrowband signal path is used.

"Wideband" The wideband signal path is used. With this setting, the dynamic range for EVM measurements is increased.

This function is only available under the following conditions:

- Instrument models FSW50/67/85
- One of the following bandwidth extension options is installed:
  – R&S FSW-B1200
  – R&S FSW-B2001
  – R&S FSW-B800R
- An I/Q bandwidth between 80 MHz and 512 MHz is used.
- The center frequency is higher than 43.5 GHz.
- The optional "Digital Baseband" interface is not active.

Remote command:
[SENSe:]IQ:WBANd on page 244

Using Electronic Attenuation
If the (optional) Electronic Attenuation hardware is installed on the FSW, 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.

For an active external frontend, electronic attenuation is not available.

Note: Electronic attenuation is not available for stop frequencies (or center frequencies in zero span) above 15 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 can 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.

For the FSW85, the mechanical attenuation can be varied only in 10 dB steps.

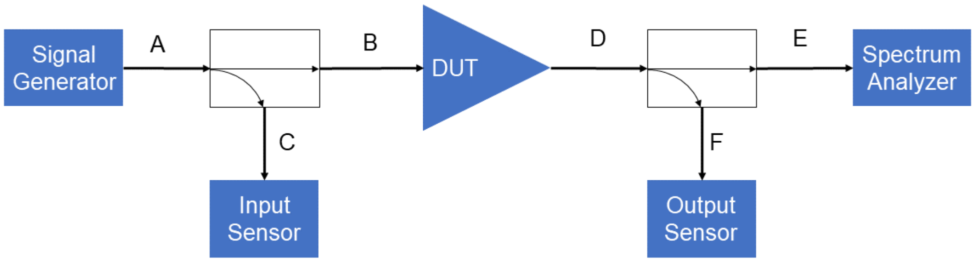
If the defined reference level cannot be set for the given attenuation, the reference level is adjusted accordingly and the warning "limit reached" is displayed in the status bar.

Remote command:
`INPut:EATT:STATe` on page 242
`INPut:EATT:AUTO` on page 241
`INPut:EATT` on page 241

3.4.4 Power sensors

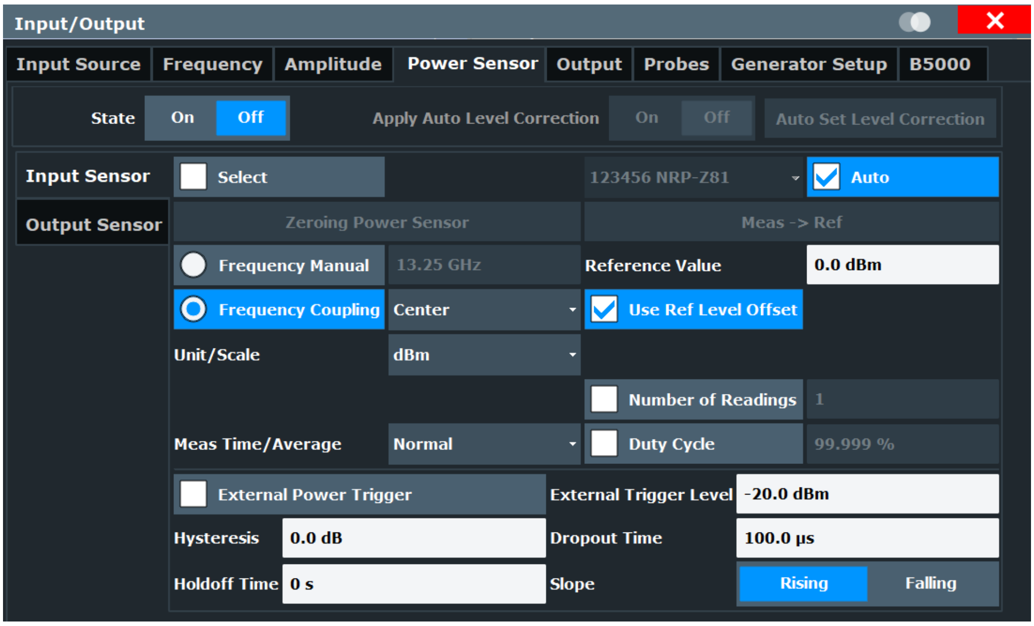
Access: "Overview" > "Input" > "Power Sensor" tab

A typical measurement using power sensors in the FSW-K18 application is set up as shown below:



For details on working with power sensors, see the FSW User Manual.

Input and output sensors are configured individually on a separate tab.



State

Switches the power measurement for all power sensors on or off. Note that in addition to this general setting, each power sensor can be activated or deactivated individually by the Select setting on each tab. However, the general setting overrides the individual settings.

Apply Auto Level Correction

This function is available after using the "Auto Set Level Correction" feature and is turned on automatically afterwards.

The measured power values from the input and output power sensors are used to correct the input and output power, respectively. This ensures that the output power measured on the I/Q data matches the output power measured by the output power sensor. The same applies to the input side.

The correction process works in the same way as the "Generator Level Offset" feature for the input power and the "Reference Level Offset" feature for the output power. However, the correction values are handled internally and are not visible, allowing the level offsets mentioned earlier to be used in addition to the level correction.

Remote command:
CALCulate:PMETer:LEVel:CORRection on page 232
[SENSe:]PMETer:LEVel:CORRection:APPLy on page 232

Select
Selects the individual power sensor for usage if power measurement is generally activated (State function).

The detected **serial numbers** of the power sensors connected to the instrument are provided in a selection list. For each of the four available power sensor indexes ("Power Sensor 1"..."Power Sensor 4"), which correspond to the tabs in the configuration dialog, one of the detected serial numbers can be assigned. The physical sensor is thus assigned to the configuration setting for the selected power sensor index.

By default, serial numbers not yet assigned are automatically assigned to the next free power sensor index for which "Auto Assignment" is selected.

Alternatively, you can assign the sensors manually by deactivating the "Auto" option and selecting a serial number from the list.

Remote command:
[SENSe:]PMETer<p>[:STATe] on page 232
SYSTem:COMMunicate:RDEVice:PMETer<p>:DEFine on page 226
SYSTem:COMMunicate:RDEVice:PMETer<p>:CONFigure:AUTO[:STATe]
on page 226
SYSTem:COMMunicate:RDEVice:PMETer<p>:COUNt? on page 226

Zeroing Power Sensor
Starts zeroing of the power sensor.

For details on the zeroing process refer to the FSW User Manual.

Remote command:
CALibration:PMETer<p>:ZERO:AUTO ONCE on page 227

Frequency Manual
Defines the frequency of the signal to be measured. The power sensor has a memory with frequency-dependent correction factors. This allows extreme accuracy for signals of a known frequency.

Remote command:
[SENSe:]PMETer<p>:FREQuency on page 229

Frequency Coupling
Selects the coupling option. The frequency can be coupled automatically to the center frequency of the instrument or to the frequency of marker 1.

Remote command:
[SENSe:]PMETer<p>:FREQuency:LINK on page 230

Unit/Scale
Selects the unit with which the measured power is to be displayed. Available units are dBm, dB, W and %.

If dB or % is selected, the display is relative to the reference value that is defined with either the "Meas -> Ref" setting or the "Reference Value" setting.

Remote command:
UNIT<n>:PMETer<p>:POWer on page 233
UNIT<n>:PMETer<p>:POWer:RATio on page 234

Meas Time/Average
Selects the measurement time or switches to manual averaging mode. In general, results are more precise with longer measurement times. The following settings are recommended for different signal types to obtain stable and precise results:

"Short"	Stationary signals with high power (> -40dBm), because they require only a short measurement time and short measurement time provides the highest repetition rates.
"Normal"	Signals with lower power or modulated signals
"Long"	Signals at the lower end of the measurement range (<-50 dBm) or Signals with lower power to minimize the influence of noise
"Manual"	Manual averaging mode. The average count is set with the Average Count (Number of Readings) setting.

Remote command:
[SENSe:]PMETer<p>:MTIMe on page 230
[SENSe:]PMETer<p>:MTIMe:AVERage[:STATe] on page 231

Setting the Reference Level from the Measurement Meas -> Ref
Sets the currently measured power as a reference value for the relative display. The reference value can also be set manually via the Reference Value setting.

Remote command:
CALCulate<n>:PMETer<p>:RELative[:MAGNitude]:AUTO ONCE on page 227

Reference Value
Defines the reference value in dBm used for relative power meter measurements.

Remote command:
CALCulate<n>:PMETer<p>:RELative[:MAGNitude] on page 227

Use Ref Level Offset
If deactivated, takes the Sensor Level Offset into account.

Remote command:
[SENSe:]PMETer<p>:ROFFset[:STATe] on page 231

Sensor Level Offset

Takes the specified offset into account for the measured power. Only available if Use Ref Level Offset is disabled.

Remote command:
`[SENSe:]PMETer<p>:SOFFset` on page 232

Average Count (Number of Readings)

Defines the number of readings (averages) to be performed after a single sweep has been started. This setting is only available if manual averaging is selected (Meas Time/Average setting).

The values for the average count range from 0 to 256 in binary steps (1, 2, 4, 8, …). For average count = 0 or 1, one reading is performed. The general averaging and sweep count for the trace are independent from this setting.

Results become more stable with extended average, particularly if signals with low power are measured. This setting can be used to minimize the influence of noise in the power sensor measurement.

Remote command:
`[SENSe:]PMETer<p>:MTIMe:AVERage:COUNt` on page 230

Duty Cycle

Sets the duty cycle to a percent value for the correction of pulse-modulated signals and activates the duty cycle correction. With the correction activated, the sensor calculates the signal pulse power from this value and the mean power.

Remote command:
`[SENSe:]PMETer<p>:DCYCle[:STATe]` on page 229
`[SENSe:]PMETer<p>:DCYCle:VALue` on page 229

Using the power sensor as an external trigger

If activated, the power sensor creates a trigger signal when a power higher than the defined "External Trigger Level" is measured. This trigger signal can be used as an external power trigger by the FSW.

This setting is only available in conjunction with a compatible power sensor.

Remote command:
`[SENSe:]PMETer<p>:TRIGger[:STATe]` on page 236
`TRIG:SOUR PSE`, see `TRIGger[:SEQuence]:SOURce` on page 233

External Trigger Level ← Using the power sensor as an external trigger

Defines the trigger level for the power sensor trigger.

For details on supported trigger levels, see the specifications document.

Remote command:
`[SENSe:]PMETer<p>:TRIGger:LEVel` on page 235

Hysteresis ← Using the power sensor as an external trigger

Defines the distance in dB to the trigger level that the trigger source must exceed before a trigger event occurs. Setting a hysteresis avoids unwanted trigger events caused by noise oscillation around the trigger level.

Remote command:
[SENSe:]PMETer<p>:TRIGger:HYSTeresis on page 235

Trigger Holdoff ← Using the power sensor as an external trigger
Defines the minimum time (in seconds) that must pass between two trigger events. Trigger events that occur during the holdoff time are ignored.

Remote command:
[SENSe:]PMETer<p>:TRIGger:HOLDoff on page 234

Drop-Out Time ← Using the power sensor as an external trigger
Defines the time the input signal must stay below the trigger level before triggering again.

Slope ← Using the power sensor as an external trigger
Defines whether triggering occurs when the signal rises to the trigger level or falls down to it.

Remote command:
[SENSe:]PMETer<p>:TRIGger:SLOPe on page 236

3.4.5 Using probes

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

The "Probes" tab of the "Input / Output" dialog box contains settings to configure probes.

Probes are a mandatory part of the test setup if you want to measure the supply voltage and the current drawn by the PA.

The probes settings are the same as those available in the spectrum application. For a comprehensive description of these settings, refer to the FSW user manual.

3.4.6 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 FSW.

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

3.4.7 Controlling a signal generator

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

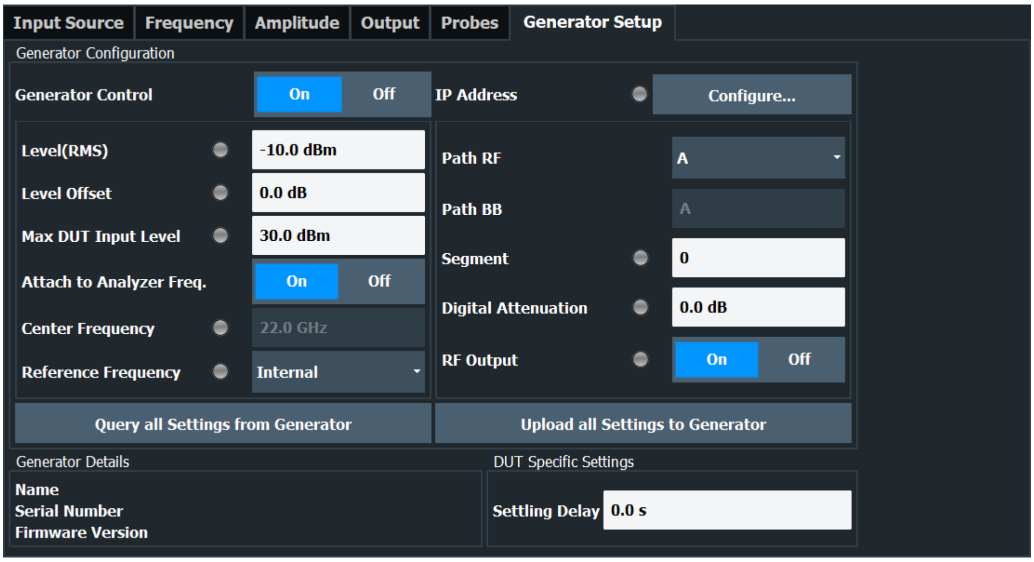
The "Generator Setup" tab of the "Input / Output" dialog box contains settings to control the signal generator from within the R&S FSW-K18. A remote control connection between the FSW and the signal generator has to be established to be able to do so.

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.

While generator control is active, you cannot change the connection information.

When you switch on generator control while it is still active in another channel, the control is disabled in the other channel. Only one channel can control a generator at any time.

Exception: The SCPI Recorder maintains control of the generator even if you switch channels.



The remote commands required to configure the generator are described in Chapter 5.6.6, "Controlling a signal generator", on page 245.

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 FSW-K18. To restore the original settings defined within the R&S FSW-K18, use that button to restore the generator settings.

Remote command:

CONFigure:GENerator:SETTings:UPDate on page 252

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 FSW.

Remote command:

CONFigure:SETTings on page 253

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

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

When a new measurement channel or save set is opened, the generator control is automatically set to "Off".

Remote command:
CONFigure:GENerator:CONTrol[:STATe] on page 246

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 FSW shows if the connection state, and, if the connection was successful, the connected generator type.

Remote command:
`CONFigure:GENerator:IPConnection:ADDRess` on page 248
`CONFigure:GENerator:CONNection:CSTate?` on page 249

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

Remote command:
RMS level: `CONFigure:GENerator:POWer:LEVel` on page 249
`CONFigure:GENerator:POWer:LEVel:LEDState?` on page 250
Level offset: `CONFigure:GENerator:POWer:LEVel:OFFSet` on page 250
`CONFigure:GENerator:POWer:LEVel:OFFSet:LEDState?` on page 251

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 246
`CONFigure:GENerator:DUT:INPut:MAXimum:POWer:LEDState?` on page 246

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 248

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 247
`CONFigure:GENerator:FREQuency:CENTer:LEDState?` on page 248

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 FSW.

Remote command:
`CONFigure:GENerator:EXTernal:ROSCillator` on page 247
`CONFigure:GENerator:EXTernal:ROSCillator:LEDState?` on page 247

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 253
BB path: `CONFigure:GENerator:TARGet:PATH:BB?` on page 253

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 252
`CONFigure:GENerator:SEGMent:LEDState?` on page 252

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 249

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 251
`CONFigure:GENerator:RFOutput:LEDState?` on page 251

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 FSW 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 245

3.4.8 Configuring the 2 GHz or 5 GHz bandwidth extension

Access: "Overview" > "Input / Frontend" > "B2000"

Access: "Overview" > "Input / Frontend" > "B5000"

The optional 2 GHz and 5 GHz bandwidth extensions allow you to capture signals with a bandwidth of up to 2 GHz and 5 GHz respectively.

Processing the data with this bandwidth requires a Rohde & Schwarz oscilloscope in the measurement setup (for example an R&S RTO). The FSW provides the signal to the oscilloscope at a (fixed) IF via an additional connector. The oscilloscope samples the signal at a rate of 10 or 20 Gigasamples, using an external frequency reference. The A/D converted data is then sent to the FSW, where it is equalized and resampled to the sample rate required by the FSW measurement application. The entire measurement and both instruments are controlled by the FSW.

For a comprehensive description on how to use the bandwidth extension, refer to the I/Q analyzer manual available for the FSW.

3.4.9 Reference: I/Q file input

3.4.9.1 Basics on input from I/Q data files

The I/Q data to be evaluated in a particular FSW application cannot 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 AM/FM/PM Modulation Analysis application.

The I/Q data file must be in one of the following supported formats:

- `.iq.tar`
- `.iqw`
- `.csv`
- `.mat`
- `.wv`
- `.aid`

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

When importing data from an I/Q data file using the import functions provided by some FSW applications, the data is only stored temporarily in the capture buffer. It overwrites the current measurement data and is in turn overwritten by a new measurement. If you use an I/Q data file as input, the stored I/Q data remains available 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, to perform measurements on an extract of the available data (from the beginning of the file) only.

For input files that contain multiple data streams from different channels, you can define which data stream to be used for the currently selected channel in the input settings. You can define whether the data stream is used only once, or repeatedly, to create a larger amount of input data.

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 FSW VSA application (R&S FSW-K70), some sample `iq.tar` files are provided in the `C:\R_S\INSTR\USER\vsa\DemoSignals` directory on the FSW.

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.

3.4.9.2 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.

Sample iq-tar files

Some sample `iq-tar` files are provided in the `C:\R_S\INSTR\USER\Demo\` directory on the FSW.

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>FSW</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

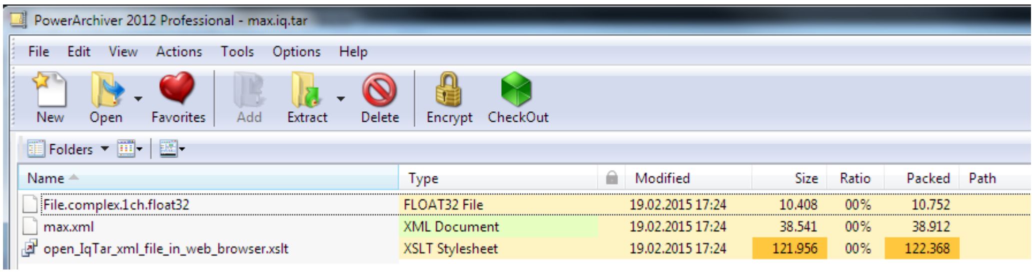
The following data elements are the minimum required for a valid iq-tar file. They are always provided by an iq-tar file export from a Rohde & Schwarz product. If not specified otherwise, it must be available in all iq-tar files used to import data to a Rohde & Schwarz product.

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).
<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.
<Clock>	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
<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 92). 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)

Element	Possible Values	Description
`<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 92). If the `<NumberOfChannels>` element is not defined, one channel is assumed.
`<DataFilename>`		Contains the filename of the I/Q data binary file that is part of the `iq-tar` file. 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) Examples: • xyz.complex.1ch.float32 • xyz.polar.1ch.float64 • xyz.real.1ch.int16 • xyz.complex.16ch.int8
`<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. FSW). 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. 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

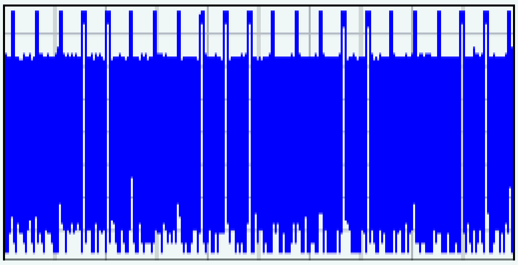
max.xml (of .iq.tar file)

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

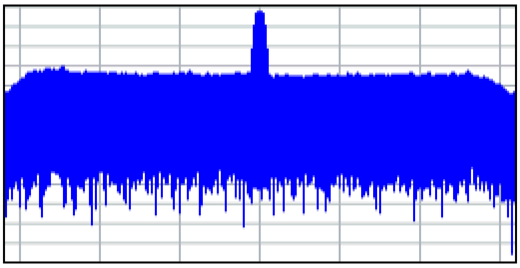
Power vs time

y-axis: 10 dB /div
x-axis: 10 ms /div

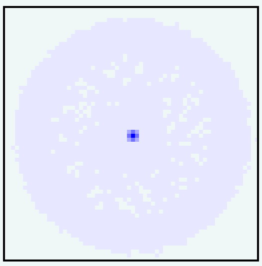


Spectrum

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>
```

3.5 Triggering

Access: "Overview" > "Trigger"

The R&S FSW-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 FSW user manual.

Note that in parallel baseband mode, you can only use trigger connector 3, the others are used to capture baseband data. When you are using an external bandwidth extension, you can only use "External Ch3" on the oscilloscope as the trigger source.

Correct TPIS

If enabled, the trigger event is set to the actual trigger point within the sample (TPIS). See also `TRACe:IQ:TPISample?` on page 297.

This setting is available for all trigger sources except "Free Run".

TPIS correction is not supported if user frequency response correction (R&S FSW-K544) is active. If you activate frequency response correction, TPIS correction is automatically deactivated in all channels.

For details see frequency response correction in the FSW user manual.

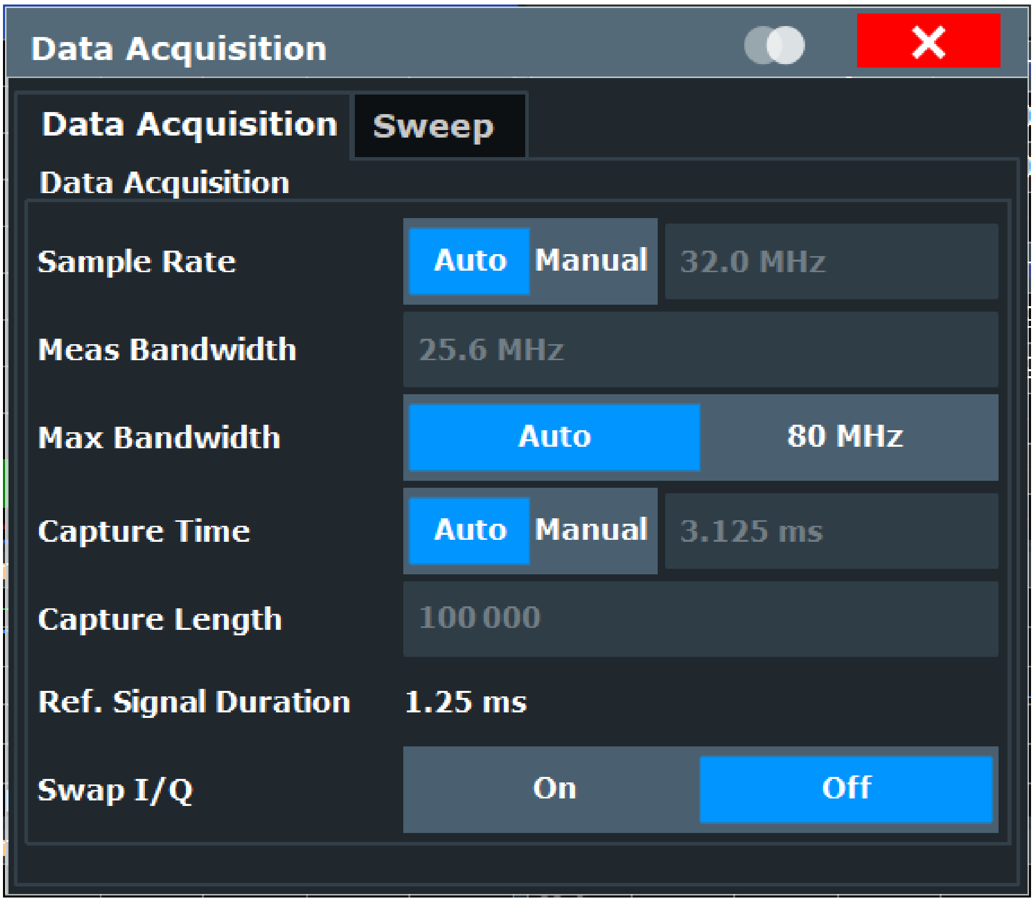
Note that when you enable this setting, the displayed trace no longer matches the instrument settings. Thus, the yellow star icon is displayed on the tab label. Start a new measurement for the correction to take effect and to update the display.

Remote command:
`TRIGger[:SEQuence]:CTPis` on page 296

3.6 Data acquisition

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 5.6.7, "Configuring the data capture", on page 254.

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 3.3, "Reference signals", on page 48.

Remote command:
Mode: `TRACe:IQ:SRATe:AUTO` on page 258

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.

Note that when you upload a new input file, the Amplifier measurement application maintains the manual sample rate definition.

Remote command:
Sample Rate: `TRACe:IQ:SRATe` on page 257
Bandwidth: `TRACe:IQ:BWIDth` on page 257

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.

When you are using the baseband input FSW-B71, the maximum bandwidth is always limited to 80 MHz.

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 FSW I/Q Analyzer.

Remote command:
`TRACe:IQ:WBANd[:STATe]` on page 259
`TRACe:IQ:WBANd:MBWidth` on page 258

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 3.3, "Reference signals", on page 48.

Remote command:
Mode: `[SENSe<ip>:]SWEep:TIME:AUTO` on page 257
Reference signal: `[SENSe:]REFSig:TIME?` on page 255

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.

When you upload a new input file, the Amplifier measurement application maintains the manual capture time definition.

Remote command:
Time: `[SENSe<ip>:]SWEep:TIME` on page 256
Capture length: `[SENSe:]SWEep:LENGth` on page 256

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.

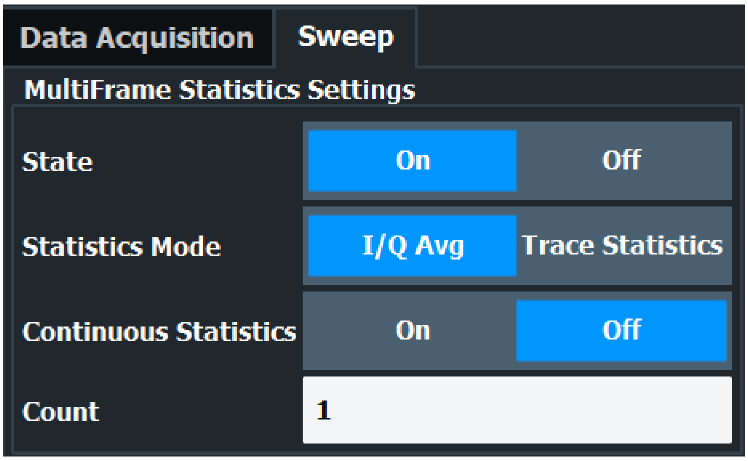
Note that the sideband is also inverted when you turn on this feature.

Remote command:
`[SENSe:]SWAPiq` on page 256

3.7 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 5.6.8, "Sweep configuration", on page 259.

Statistics State

Turns the sweep statistics calculation on and off.

Remote command:
`[SENSe:]SWEep:STATistics[:STATe]` on page 261

Statistics Mode

Sets the statistics mode.

If I/Q averaging is selected, the IQ data is averaged over several data captures after synchronization to the reference file. This leads to a significant noise reduction. Be aware that 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.

If trace statistics is selected, multiple frames are measured to create a graphical or scalar statistics result.

Remote command:
`[SENSe:]SWEep:STATistics:MODE` on page 261

Continuous Statistics

If continuous statistics is enabled, it does not reset the results when the average count is through. Instead, it continues to average the data. The continuous statistics setting only has an effect in continuous sweep mode.

If continuous statistics is turned off, averaging starts again from "0" after the defined statistics count is reached.

Remote command:
[SENSe:]SWEep:STATistics:CONTinuous[:STATe] on page 261

Statistics Count
Defines the number of single data captures the application uses to average the data.

Remote command:
[SENSe:]SWEep:STATistics:COUNt on page 261

Select Result Rng
Sets the result range.

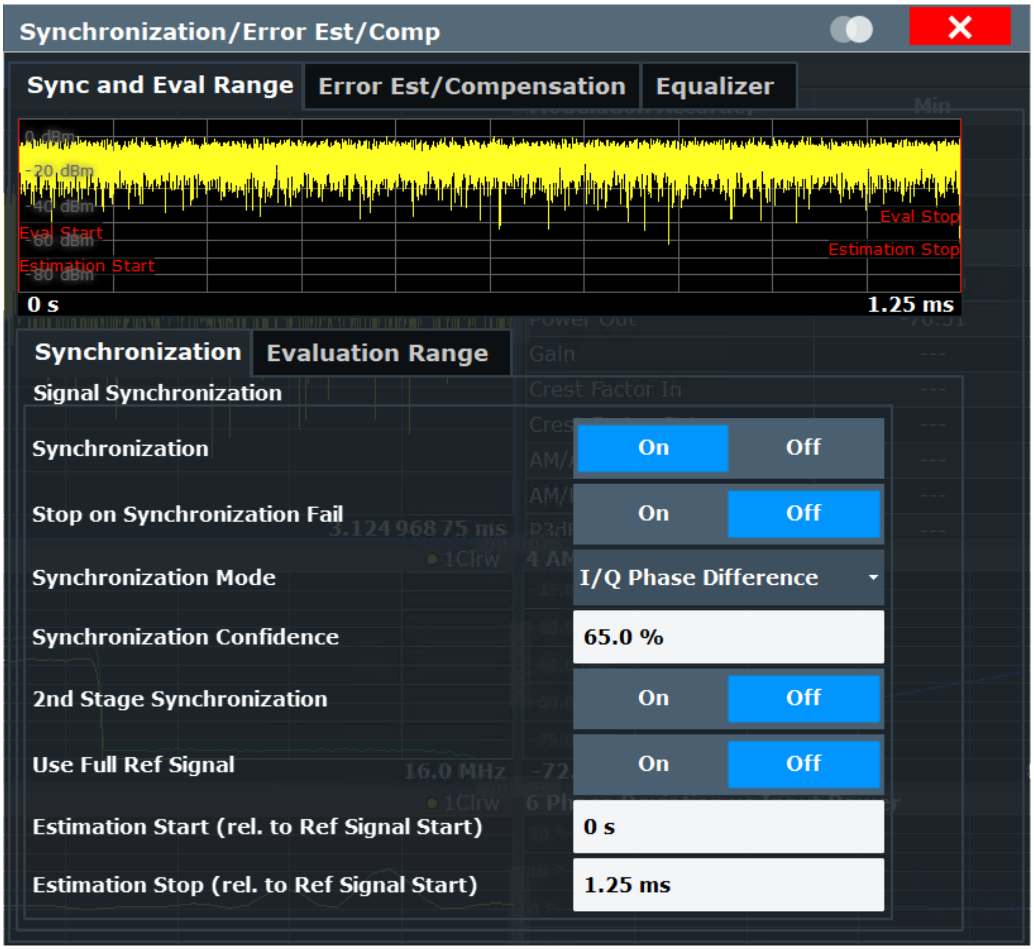
Remote command:
CONFigure:RESult:RANGe[:SELected] on page 261

3.8 Synchronization

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 5.6.9, "Synchronizing measurement data", on page 262.

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).

Remote command:
CONFigure:SYNC:STAT on page 264

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 264

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 263

2nd Stage Synchronization
The second stage synchronization activates an additional synchronization algorithm (operating in frequency domain).

The activation is recommended especially for e.g. two carrier signals with a large spacing between the carriers, when the standard synchronization delivers unstable results.

As it is an additional synchronization step, it increases the measurement time.

Remote command:
`CONFigure:SYNC:SECond:STAT` on page 264

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.

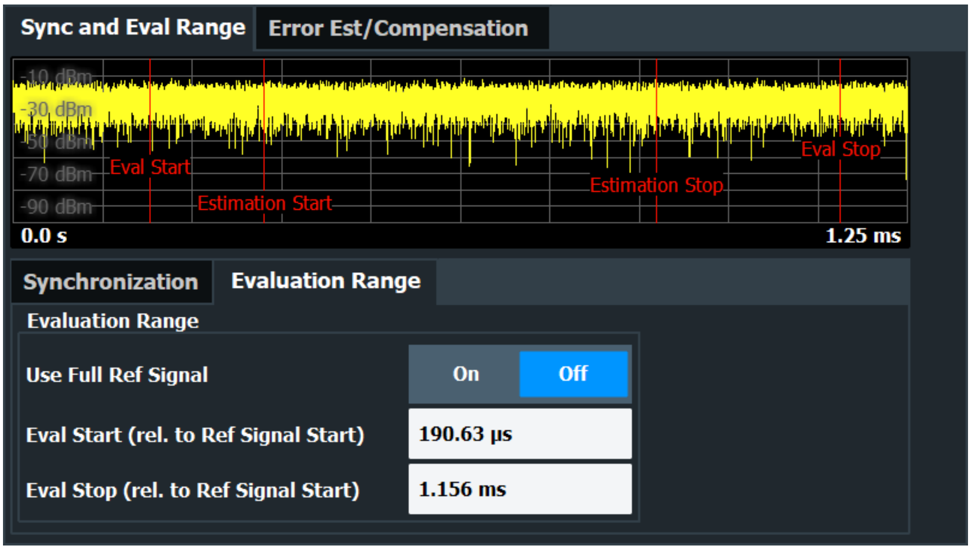
Tip: You can also use the touchscreen to move the lines to a new position in the preview pane. However, this way is not as accurate as entering a number into the input field.

Remote command:
`CONFigure:ESTimation:FULL` on page 262
`CONFigure:ESTimation:STARt` on page 263
`CONFigure:ESTimation:STOP` on page 263

3.9 Evaluation range

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 5.6.10, "Defining the evaluation range", on page 265.

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.

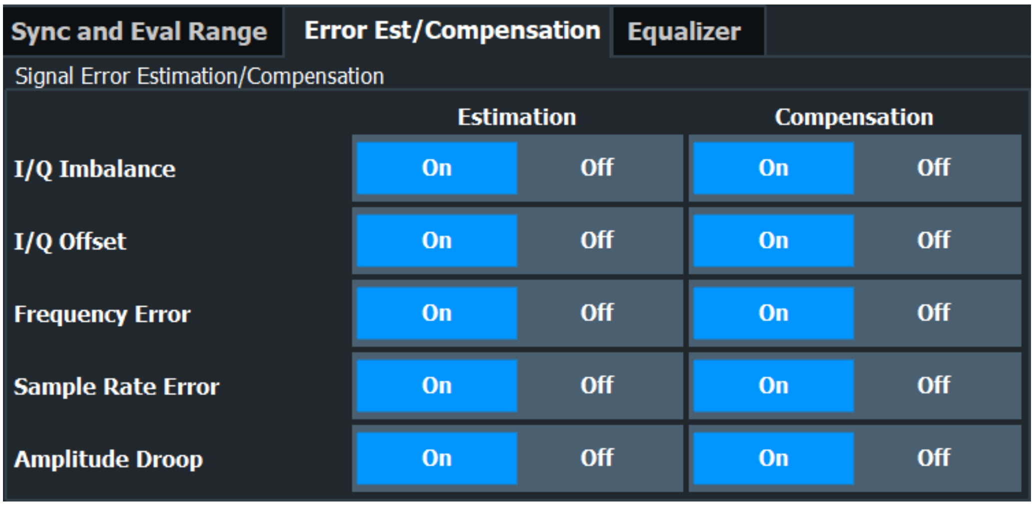
Tip: You can also use the touchscreen to move the lines to a new position in the preview pane. However, this way is not as accurate as entering a number into the input field.

Remote command:
`CONFigure:EVALuation:FULL` on page 265
`CONFigure:EVALuation:RANGe` on page 266
`CONFigure:EVALuation:STARt` on page 266
`CONFigure:EVALuation:STOP` on page 267

3.10 Signal error estimation and compensation

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 5.6.11, "Estimating and compensating signal errors", on page 267.

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.

Remote command:
`CONFigure:SIGNal:ERRor:ESTimation:IQIMbalance[:STATe]` on page 269
`CONFigure:SIGNal:ERRor:COMPensation:IQIMbalance[:STATe]` on page 268

I/Q Offset
Shift of the constellation points in a particular direction.

Remote command:
`CONFigure:SIGNal:ERRor:ESTimation:IQOFfset[:STATe]` on page 270
`CONFigure:SIGNal:ERRor:COMPensation:IQOFfset[:STATe]` on page 268

Frequency Error
Difference between measured and reference center frequency.

Remote command:
`CONFigure:SIGNal:ERRor:ESTimation:FERRor[:STATe]` on page 269
`CONFigure:SIGNal:ERRor:COMPensation:FERRor[:STATe]` on page 268

Sample Error Rate
Difference between the sample rate of the reference signal and the measured signal.

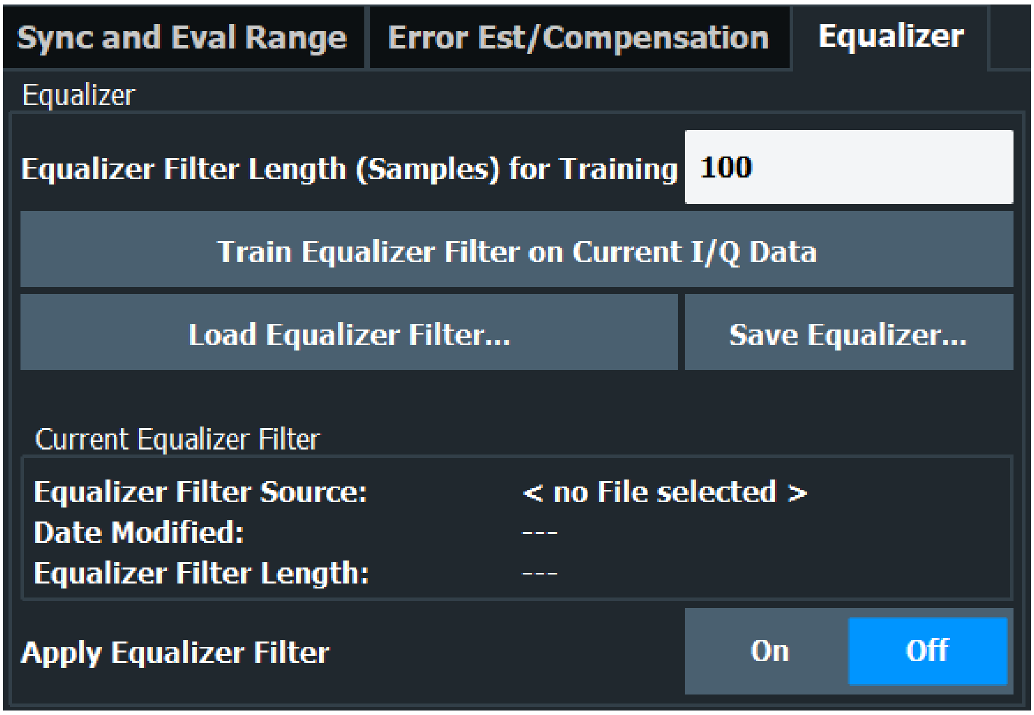
Remote command:
`CONFigure:SIGNal:ERRor:ESTimation:SRATe[:STATe]` on page 270
`CONFigure:SIGNal:ERRor:COMPensation:SRATe[:STATe]` on page 269

Amplitude Droop
Decrease of the signal power over time in the transmitter.

Remote command:
`CONFigure:SIGNal:ERRor:ESTimation:ADRoop[:STATe]` on page 269
`CONFigure:SIGNal:ERRor:COMPensation:ADRoop[:STATe]` on page 267

3.11 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 FSW, or use one that you already have.

Training (or creating) the equalizer filter is a process in which the FSW 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 FSW 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 FSW 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 FSW 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.

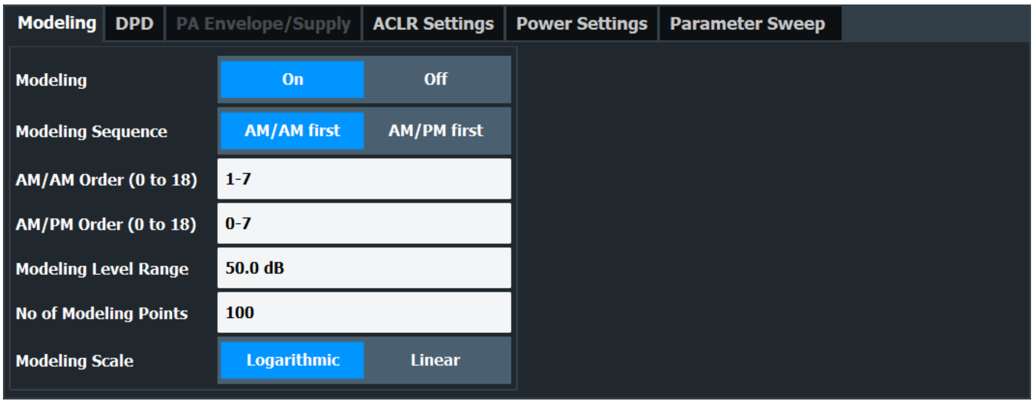
`TRACe:IQ:EQUalized?` on page 406

Remote command:
Filter length: `CONFigure:EQUalizer:FILTer:LENGth` on page 271
Start training: `CONFigure:EQUalizer:TRAin` on page 272
Store filter: `MMEMory:STORe<n>:EQUalizer:FILTer:COEFficient` on page 272
File format: `CONFigure:EQUalizer:FILTer:FILE:FORMat` on page 270
Restore filter: `MMEMory:LOAD:EQUalizer:FILTer:COEFficient` on page 272
Equalizer state: `CONFigure:EQUalizer[:STATe]` on page 271
Manual filter definition: `CONFigure:EQUalizer:FPARameters` on page 271

3.12 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 5.6.12, "Applying a system model", on page 273.

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 FSW-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 275
`CONFigure:MODeling:SEQuence` on page 274

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 273
"AM/PM": `CONFigure:MODeling:AMPM:ORDer` on page 273

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 274
Points: `CONFigure:MODeling:NPOints` on page 274

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 274

3.13 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 FSW 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 FSW 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 3.4.7, "Controlling a signal generator", on page 79.

Note:

When you create a `.wv` waveform file using digital predistortion, it only applies to the specific generator level and generator attenuation that were set during creation. The values used during creation can be seen in the `.wv` file header comment, for example: "{COMMENT: Created by Rohde & Schwarz FS-K18 Amplifier Measurement application. Signal generated for SMx RMS level: -6.8 dBm and with SMx attenuation 0 dB.}". Create a new waveform file if you have changed the generator level or generator attenuation.

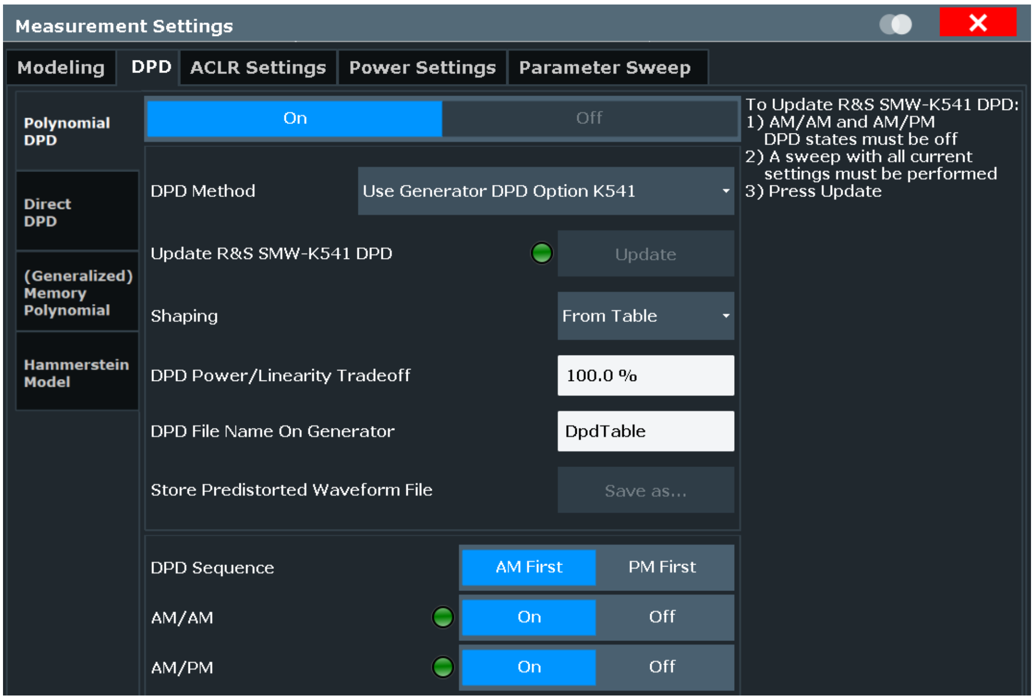
Remote command:

`CONFigure:DDPD[:STATe]` on page 286

3.13.1 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 FSW 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 5.6.13, "Applying digital predistortion", on page 275.

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 FSW with the "Update" button. (The button is only available when data has been captured on the FSW 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 FSW 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 278

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

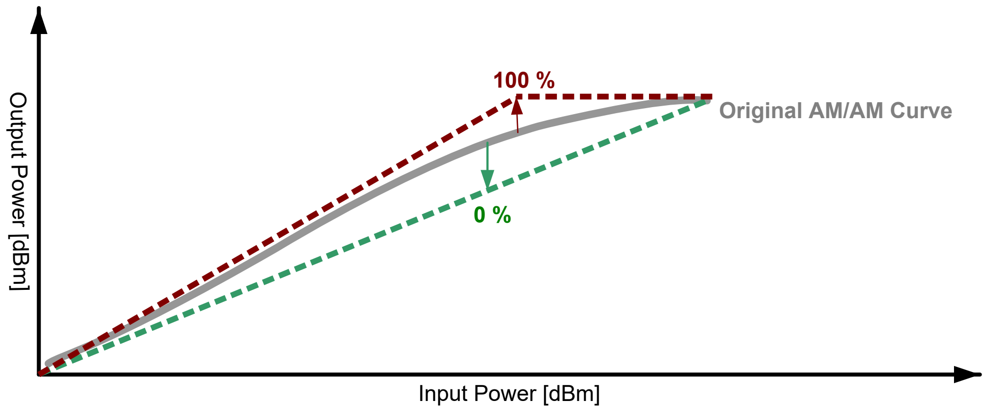
- "From Table"
  Shapes the DPD function based on a table that contains the correction values required to predistort the signal.
  The calculation of the table is based on the "AM/AM" and "AM/PM" polynomial models.
  For more information about the contents and usage of the shaping table, refer to the 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 3.12, "System models", on page 108.

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

Remote command:
Mode: `CONFigure:DPD:SHAPing:MODE` on page 280
Table name: `CONFigure:DPD:FNAMe` on page 278

Polynomial 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 ideal case (red line), the DPD affects the amplifier characteristics in a way that the best output power is achieved.



Remote command:
`CONFigure:DPD:TRADeoff` on page 280

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 276
"AM/PM" state: `CONFigure:DPD:AMPM[:STATe]` on page 277
Both: `CONFigure:DPD:AMXM[:STATe]` on page 277
Calculation order: `CONFigure:DPD:SEQuence` on page 279

3.13.2 Direct DPD (FSW-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 FSW applies the correction values, measures the improved input signal again, applies the correction values etc. This process goes on until the number of iterations that 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 if synchronization is not possible during direct DPD, FSW-K18 continues with a new measurement (including capture) until synchronization was successful. 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.

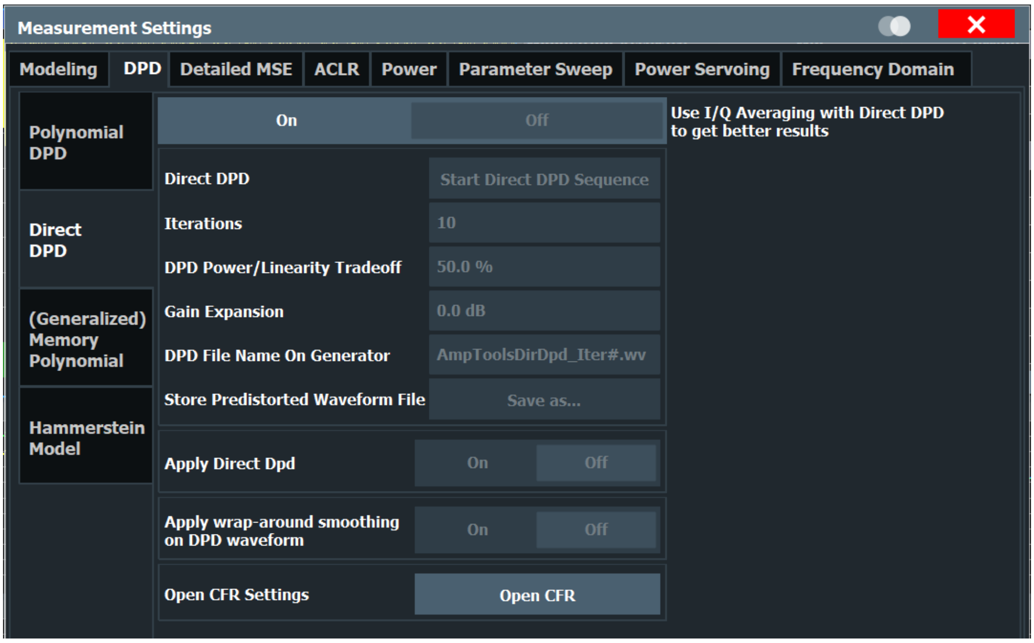
Continuous statistics during direct DPD calculation

Continuous statistics is automatically disabled during the direct DPD calculation.

Generator control during direct DPD calculation

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

Also, the I/Q modulator of the signal generator is set to high quality table mode.



The remote commands required to configure the direct DPD are described in Chapter 5.6.13, "Applying digital predistortion", on page 275.

The "Open CFR" button opens the "Crest Factor Reduction (K548 Control)" dialog of the "Reference Signal" configuration.

Automated 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 FSW 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 FSW 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 "#" in the file name will be replaced by the iteration number. 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.)

"Apply wrap-around smoothing on DPD waveform" smoothes start- and tail-samples down to "0" in order to avoid phase discontinuities when the file is cyclically played from a signal source.

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 anytime 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 284
Start sequence: `CONFigure:DDPD:STARt` on page 286
Gain expansion: `CONFigure:DDPD:GEXPansion` on page 286
File name: `CONFigure:DDPD:FNAMe` on page 285
Save DPD: `MMEMory:STORe<n>:DDPD` on page 288
Apply DPD: `CONFigure:DDPD:APPLy[:STATe]` on page 283
Wrap-around smoothing: `CONFigure:DDPD:APPLy:WRAP[:STATe]` on page 284
Query I/Q values: `TRACe:IQ:DDPD[:DATA]?` on page 319

Manual direct DPD sequence

The direct DPD method requires one or more measurements (or iterations) to determine the correction values. The manual direct DPD sequence described here can be used, for example, to perform measurements on RF simulations and Digital2RF devices like fully integrated frontends containing an amplifier. When generator control is off, manual direct DPD mode is activated automatically.

When you select the "Start Direct DPD Sequence" button, the FSW runs the first iteration to calculate DPD.

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.

While the calculation is running, you can "Abort" the calculation. This stops the DPD sequence and discards the predistorted I/Q data that have already been calculated.

When the calculation is finished, a path to a `.wv` file with the calculated DPD values for the current iteration is displayed. Upload this file to the signal source and set the level on the source to the value saved in the `RMSLevel.txt` in the same location.

After you have uploaded the waveform file and set the level on the signal source, press "Continue" to run another iteration based on the new waveform file. A new `.wv` file is created with each iteration and another level value is added to the `RMSLevel.txt` document.

The process ends automatically when you have reached the defined number of iterations. If you want to end the process earlier, select "Finish" to end the DPD sequence.

You can also 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.

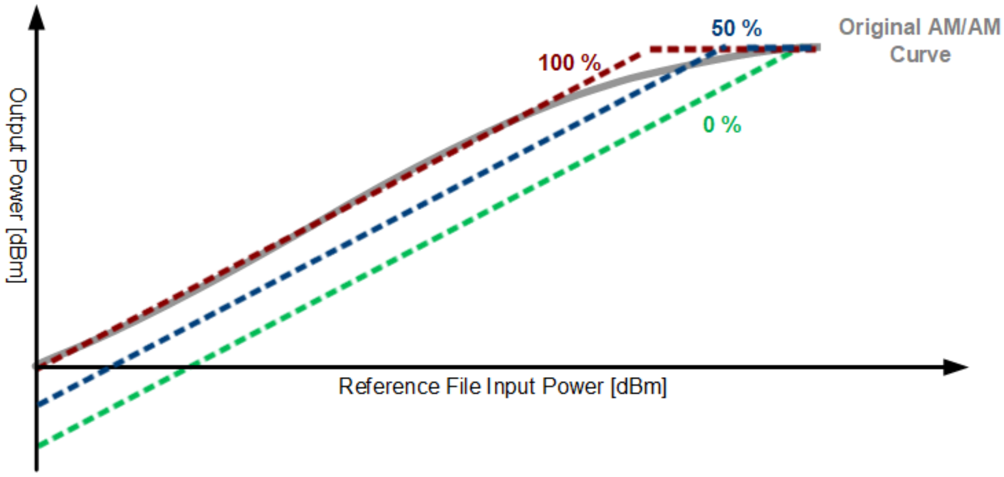
"Apply wrap-around smoothing on DPD waveform" smoothes start- and tail-samples down to "0" in order to avoid phase discontinuities when the file is cyclically played from a signal source.

Remote command:
Iterations: `CONFigure:DDPD:COUNt` on page 284
Query finished iterations: `CONFigure:DDPD:COUNt:CURRent?` on page 285
Start sequence: `CONFigure:DDPD:STARt` on page 286
Gain expansion: `CONFigure:DDPD:GEXPansion` on page 286
Wrap-around smoothing: `CONFigure:DDPD:APPLy:WRAP[:STATe]` on page 284
Continue direct DPD: `CONFigure:DDPD:CONTinue` on page 284
Query RMS power: `CONFigure:MDPD:RMS[:CURRent]?` on page 292
Query RMS power (online mode): `CONFigure:DDPD:RMS[:CURRent]?` on page 286
Query I/Q values: `TRACe:IQ:DDPD[:DATA]?` on page 319
Query file name and path: `FETCh:DDPD:WAVeform:PATH?` on page 288

Direct 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:
CONFigure:DDPD:TRADeoff on page 287

3.13.3 (Generalized) memory polynomial DPD (FSW-K18M)

The FSW-K18M application is an extension to the FSW-K18D Direct DPD application. It is only available after a valid automated or manual direct DPD sequence is run. In FSW-K18M the application derives a (generalized) memory polynomial equation that transfers the reference signal (ideal waveform) into the pre-distorted waveform (K18D waveform).

$$y_{GMP}(n) = \sum_{k=1}^{K_a}\sum_{l=0}^{L_a} a_{kl}\cdot x(n-l)\cdot |x(n-l)|^{k-1} + \sum_{k=2}^{K_b}\sum_{l=0}^{L_b}\sum_{m=1}^{M_b} b_{klm}\cdot x(n-l)\cdot |x(n-l-m)|^{k-1} + \sum_{k=2}^{K_c}\sum_{l=0}^{L_c}\sum_{m=1}^{M_c} c_{klm}\cdot x(n-l)\cdot |x(n-l+m)|^{k-1}$$

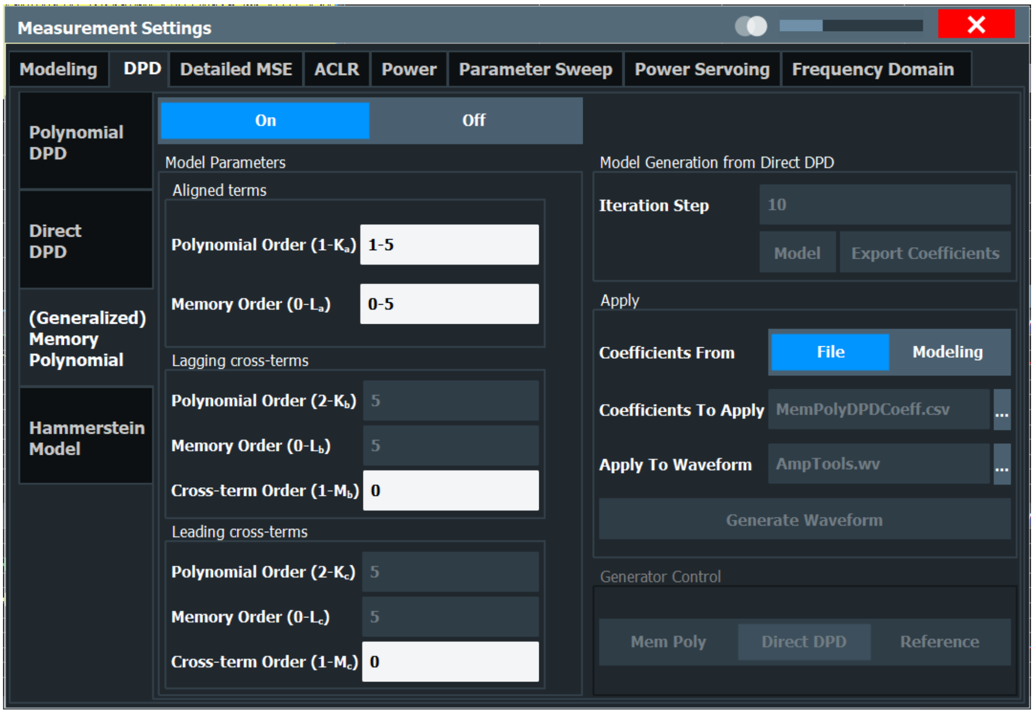
The reference signal is represented by x, scaled to volt. The coefficients are shown in the Memory DPD coefficients display, or directly exported into a `csv` file using the export coefficients function. In the Memory DPD coefficients display, only the values of the aligned terms are displayed as long as the cross-term order fields (1-M_b and 1-M_c) remain at default.

You can also apply the coefficients to the reference signal automatically and upload the resulting waveform to the generator using the export waveform function.

The results are visible in the Memory DPD Coefficients result display.

If a "Model generation failed" error occurs, it can be rectified as follows. Either reduce the reference signal length (by adjusting the evaluation range (see "Defining the evaluation range" on page 104) or using a shorter reference signal, e.g. by generating an own signal (see "Designing a reference signal within the R&S FSW-K18" on page 54). It also helps to reduce the number of coefficients.

Alternatively you can apply the coefficients to a signal by using MATLAB. For more details, see the sample implementations shown in Chapter 5.11, "Programming examples FSW-K18M", on page 409.



The remote commands required to configure the memory polynomial DPD are described in Chapter 5.6.13, "Applying digital predistortion", on page 275.

Read-in the memory polynomial model coefficients from CSV file

After exporting a `csv` file by using "Export Coefficients", it is possible to change the coefficients and read-in the `csv` file again. All values for the columns are return values of the digital signal processing. Note that the coefficients should be comma "," separated. To activate the file mode select "File". The button "Modeling" deactivates the file mode.

In file mode, the number of coefficients for modeling and the number of coefficients in the selected `csv` file are checked. If there is a warning, it is displayed only in the status bar and the "Generate Waveform" or "Generate & Export Waveform" button is grayed out.

"Apply to Waveform" only supports `wv` files.

The level that is considered for the scaling of the "Apply to Waveform" is the one that can be selected in the "Reference Signal" dialog as "RMS Power" (see Level (RMS)). For the "RMS Power", an entry is required. When the generator control is on, the level changes accordingly to the value that is set for the finished memory polynomial model waveform that is selected at the generator.

Running a Memory Polynomial DPD sequence

The memory polynomial DPD method creates an equation with a memory polynomial, where the polynomial order and memory order can be specified on the user interface.

Per default, the "Polynomial Order" of the aligned terms is 1-5, i.e. all orders from 1 to 5. If not all orders shall be taken into account, the same notation as in the modelling dialog may be used, e.g. 1-3;5 includes 1, 2, 3, and 5.

The "Memory Order" of the aligned terms is configurable in the same way and describes the number of filter taps to be used per filter. It uses zero-based indexing, as it describes the "delays", so "0" corresponds to "no filter".

The input fields for the lagging and leading cross-terms require maximum values to be entered.

Keep in mind that the computational effort for the model increases with memory order, polynomial order, and length of the waveform (in samples).

Also, a specific "Iteration Step" to be used for the modeling can be selected. Per default, this is the last iteration, but any other step can be selected as well. The DDPD result window shows ACLR and / or EVM results over iterations and helps selecting the right iteration step.

After the parameters for "Polynomial Order", "Memory Order" and "Iteration Step" have been defined, selecting "Model" starts the fitting of the memory polynomial and calculates the coefficients.

Once the modeling is complete, the coefficients are visible in the Memory DPD Coefficients result display. You can either use "Export Waveform" to export the waveform with the model applied to the generator, or select "Export Coefficients" to export the coefficients to a file. "Export Waveform" and "Export Coefficients" is only available after a "Model" has been derived.

The model is applied to the file specified under "Apply to". The default value is the current reference file - however, the model may be applied to any waveform file. It is recommended to only apply the model to signals that are similar to the reference signal used for direct DPD, especially with regard to bandwidth and crest factor. "Export Waveform" with generator control off will open a "Save to" dialog allowing export of the waveform with the memory polynomial model applied.

With the memory-polynomial waveform transferred to the generator using "Export Waveform", you can switch between the "Memory Polynomial" and the "Direct DPD" waveform to compare the pre-distortion results. Selection of DPD signal (on the signal generator) is only available after "Export Waveform" in generator control mode.

Remote command:
State: `CONFigure:MDPD[:STATe]` on page 291
Iteration Step: `CONFigure:MDPD:ITERation` on page 291
Polynomial Order: `CONFigure:MDPD:ORDer:POLYnomial` on page 291
Memory Order: `CONFigure:MDPD:ORDer:MEMory` on page 291
Apply To: `CONFigure:MDPD:APPLy:MODel` on page 290

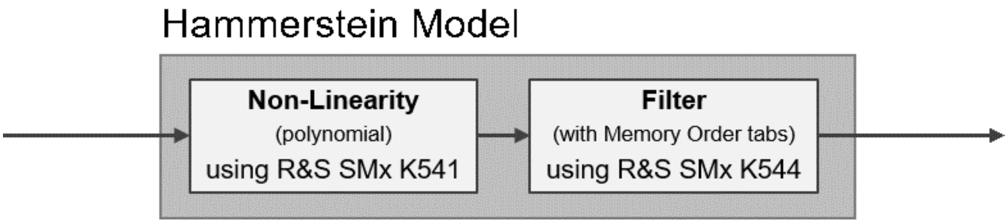
Model: CALCulate:MDPD:MODel on page 291
Export Waveform (only available when generator control is OFF): MMEMory:STORe:MDPD:WAVeform on page 293
Export Coefficients: MMEMory:STORe:MDPD:COEFficient on page 293
Waveform Type: CONFigure:MDPD:WAVeform:SELect on page 292
Send Waveform to Generator: CONFigure:MDPD:WAVeform:UPDate on page 292

3.13.4 Hammerstein model (FSW-K18M)

The Hammerstein model is a DPD approach that is, like the Memory Polynomial Model, available in FSW-K18M. Both are based on the results of the FSW-K18D Direct DPD and therefore require a valid Direct DPD result.

As the Hammerstein model is a real-time approach, i.e. the predistortion is applied in real time on the signal generator as it plays the undistorted signal, the signal generator must be equipped with options R&S SMx-K541 and K544. The Hammerstein model consists of a static nonlinearity followed by a linear filter. The FSW-K18M Hammerstein model uses an FIR filter. Due to the combination of nonlinearity and filter, the Hammerstein model can model nonlinear behavior and memory effects. The Hammerstein model can be seen as a simplification of the Memory Polynomial model, which leads to a lower complexity for realization.

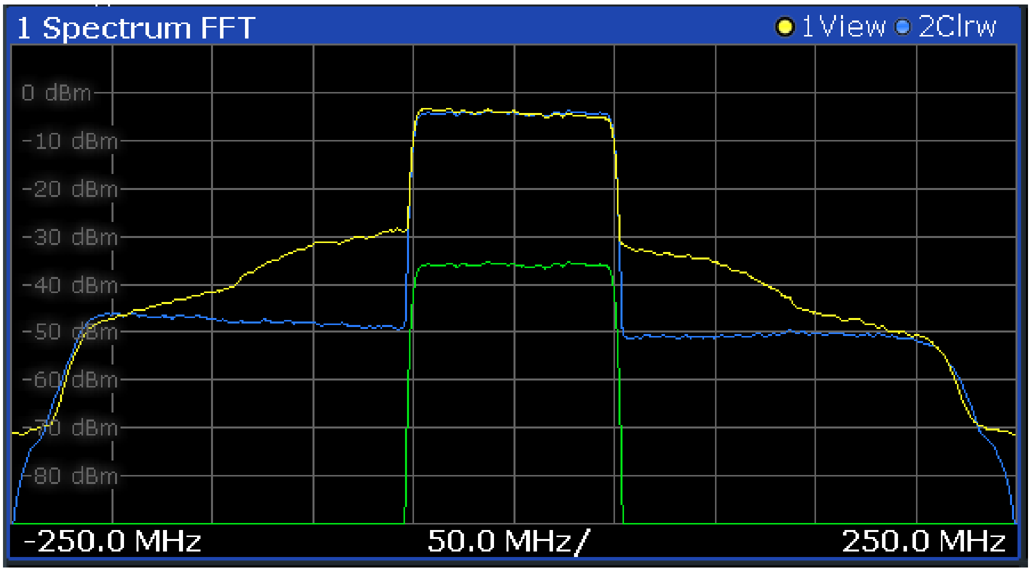
The parameters of the Hammerstein model are calculated based on the reference waveform (ideal waveform) and the pre-distorted waveform (K18D waveform), similar to the Memory Polynomial DPD. After a valid Direct DPD sequence, the Hammerstein model parameters can be computed. The Hammerstein model parameters are then used to configure R&S SMx-K541 and K544 at a connected R&S generator.



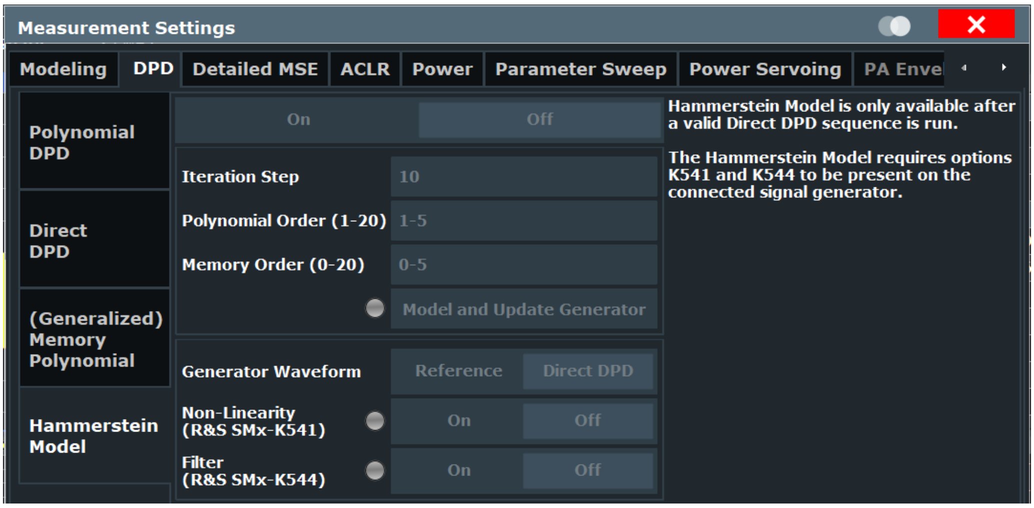
The advantage of the Hammerstein model is that by using R&S SMx-K541 and K544, the predistortion is applied in real time on the signal in an R&S generator. Real-time means that the predistortion is added to the undistorted signal by the signal generator as the undistorted signal is being created or played back. Due to its real-time applicability, a Hammerstein parameter set does not only apply to one given signal at a given level, but can be applied to a different signal or lower power levels as well.

Note that exchanging the signal while keeping the Hammerstein model parameters works only within certain boundaries, i.e. similar signal characteristics (e.g. PAPR, bandwidth). Keep in mind that a direct DPD with unlimited degrees of freedom in general results in better performance compared to any real-world model with limited degrees in freedom (e.g. polynomial degree and filter order).

For best results, it is recommended to use Direct DPD with I/Q averaging as well as an increased measurement bandwidth for better ACLR results. The I/Q averaging is not needed afterwards for the Hammerstein model. As usual for any modelling DPD - it is recommended to include all relevant out of band non-linearities into the analysis bandwidth. A well known rule of thumb is a factor of 3 to 5 times the signal bandwidth. The reason for the increased measurement bandwidth can be seen in the following figure with an increased bandwidth (factor 4).



The effect of the non-linear behavior of the DUT is not only limited to the bandwidth of the signal itself, but also affects adjacent frequencies. This leads to “shoulders” in the spectrum as can be seen for the measured signal (yellow) compared to the signal, read from file (green). These “shoulders” can be improved by the Direct DPD (blue) if they are included in the measurement by increasing the measurement bandwidth. The same holds for the Hammerstein model as it is derived from the Direct DPD result.



The remote commands required to configure the Hammerstein model are described in Chapter 5.6.13, "Applying digital predistortion", on page 275.

Running a Hammerstein model sequence
The Hammerstein model is a real-time approach, and consists of a static non-linearity followed by a linear filter.

Per default, the "Polynomial Order" is 1-5, i.e. all orders from 1 to 5. If not all orders shall be taken into account, the same notation as in the modeling dialog may be used, e.g. 1-3;5 includes 1, 2, 3, and 5.

The "Memory Order" is configurable in the same way and describes the number of filter taps to be used per filter. It uses zero-based indexing, as it describes the "delays", so "0" corresponds to "no filter".

Keep in mind that the computational effort for the model increases with memory order, polynomial order, and length of the waveform (in samples).

Also, a specific "Iteration Step" to be used for the modeling can be selected. Per default, this is the last iteration, but any other step can be selected as well. The DDPD result window shows ACLR and / or EVM results over iterations and helps selecting the right iteration step.

After the parameters for "Polynomial Order", "Memory Order" and "Iteration Step" have been defined, selecting "Model and Update Generator" starts the fitting of the model and exports the results to the signal generator.

After using "Model and Update Generator", you can compare the pre-distortion results by switching "Generator Waveform" from "Reference" to "Direct DPD". If "Reference" is selected, "Non-linearity" and "Filter" are automatically switched on and the Hammerstein Model is applied at the generator. Now the "Non-linearity" and "Filter" options can be switched on and off manually and independent from each other if needed.

Remote command:
State: `CONFigure:HAMMerstein[:STATe]` on page 294
Iteration Step: `CONFigure:HAMMerstein:ITERation` on page 294
Polynomial Order: `CONFigure:HAMMerstein:ORDer:POLYnomial` on page 294
Memory Order: `CONFigure:HAMMerstein:ORDer:MEMory` on page 294
Model and Update Generator:`CONFigure:HAMMerstein:MUPGenerator` on page 294
Generator Waveform: `CONFigure:HAMMerstein:GENWaveform[:SELect]` on page 295
Non-Linearity: `CONFigure:HAMMerstein:NONLinearity[:STATe]` on page 295
Filter: `CONFigure:HAMMerstein:FILTer[:STATe]` on page 295

3.14 Detailed MSE

Access: "Overview" > "Measurement" > "Detailed MSE"

Detailed MSE provides functionality to estimate the fraction of the error power of the three main distributors to the EVM: Noise, non-linearities and frequency response. Since the fraction of the error power is considered, the mean squared error (MSE) is measured as the squared EVM in percent.

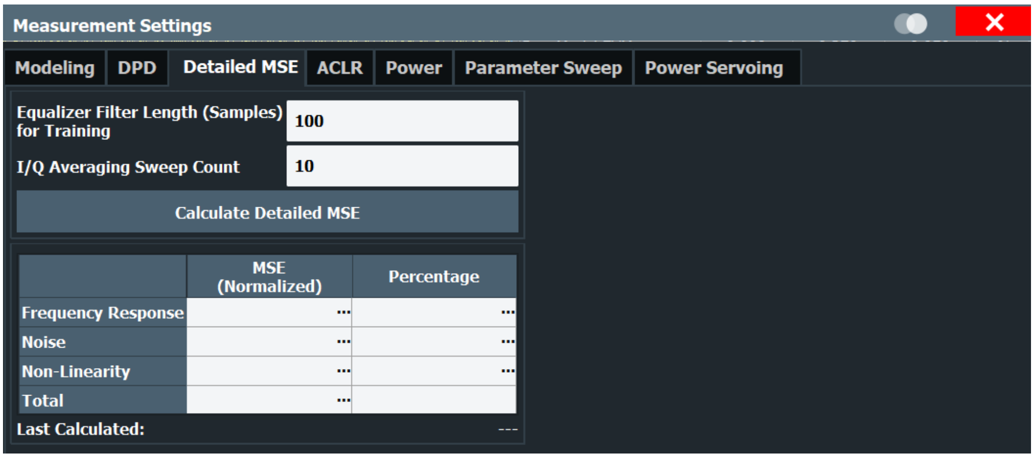
To measure the MSE values of the noise, non-linearities and the frequency response, IQ averaging and equalizer are used. The measurement sequence starts, when the button "Calculate Detailed MSE" is pressed.

MSE is normalized by the mean signal power and is calculated as follows, where "e_i" are elements of the complex error vector, "r_i" the complex amplitudes of the signal and "N" the length of the signal.

$$EVM_{in\%} = \sqrt{\frac{\frac{1}{N}\sum_{i=1}^{N}|e_i|^2}{\frac{1}{N}\sum_{i=1}^{N}|r_i|^2}} \cdot 100\%$$

$$MSE_{normalized} = EVM_{in\%}^2 = \frac{\frac{1}{N}\sum_{i=1}^{N}|e_i|^2}{\frac{1}{N}\sum_{i=1}^{N}|r_i|^2} \cdot (100\%)^2 = \frac{MSE}{Mean\ Signal\ Power} \cdot (100\%)^2$$

This functionality is a calculated estimation and not a precise measurement. It is only run if "Calculate Detailed MSE" on page 126 is executed and not updated automatically for every new measurement.



Equalizer Filter Length For Training

Defines the number of FIR filter coefficients to be calculated. A larger number of samples generally yields better results, but takes longer to calculate.

Remote command:

`CONFigure:EQUalizer:FILTer:LENGth` on page 271

I/Q Averaging Sweep Count

Defines the number of single data captures the application uses to average the data.

Remote command:

`[SENSe:]SWEep:IQAVg:COUNt` on page 259

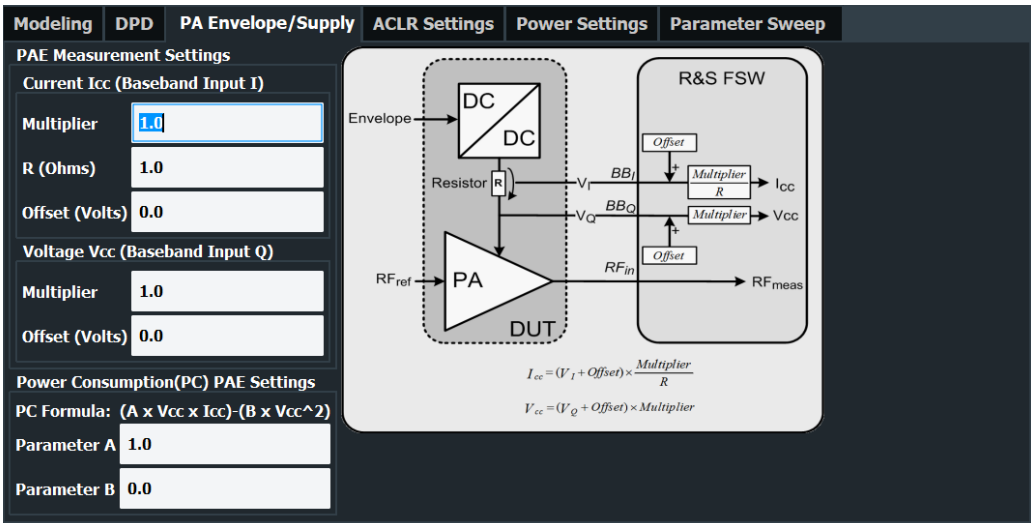
Calculate Detailed MSE
Runs the detailed MSE calculation.

Remote command:
`CALCulate:MSERror?` on page 296

3.15 Envelope measurements

Access: "Overview" > "Measurement" > "PA Envelope / Supply"

When you analyze power amplifiers supporting envelope tracking, you have to describe several characteristics of the measurement equipment to get valid results.



The remote commands required to configure the envelope tracking are described in Chapter 5.6.16, "Configuring envelope tracking", on page 298.

Configuring PAE measurements (Power Added Efficiency)
When you are testing amplifiers that support envelope tracking, the Power Added Efficiency (PAE) of the system is the value that characterizes its performance.

To calculate the PAE, you have to measure the supply voltage and current drawn by the power amplifier. The PAE is calculated according to the following equation:

PAE = (RF Output Power - RF Input Power) / DC Power

with DC Power = Voltage * Current

Measuring the voltage and current requires additional equipment and components in the test setup. For valid measurement results, you have to define the characteristics for those components.

Required components

One way to measure the voltage is to use a probe. The voltage is measured on the Q channel of the baseband input provided by the optional baseband hardware option.

One way to measure the current is to use a shunt resistor and another probe. The current is measured on the I channel of the baseband input provided by the optional baseband hardware option.

For both types of components, you have to define their characteristics and behavior accurately.

Measuring current

When using a shunt resistor to measure the current, you have to define the **resistance R** of the shunt resistor you are using. The resistance is a value with the unit Ω.

If the test setup contains additional components (for example passive probes), you can define their characteristics by specifying a **multiplier**. The multiplier is a value without unit. When you are using an active probe from Rohde & Schwarz, you do not have to change the multiplier, because it is automatically detected by the FSW.

In addition, you have to compensate the DC offset of active probes. The DC offset is described by the **offset** value, which differs depending on the probe you are using. You can figure out the offset value by measuring it.

Measuring voltage

For voltage measurements, you have to define the **multiplier** (to take the attenuation of passive probes into account) and the **offset** (to compensate the DC offset of active probes).

Note that entering wrong values for these parameters yields invalid measurement results. Generally speaking, the multiplier multiplies the results by a certain value, the offset is added to the results.

These settings are available when you turn on the baseband input.

Remote command:
See Chapter 5.6.16, "Configuring envelope tracking", on page 298.

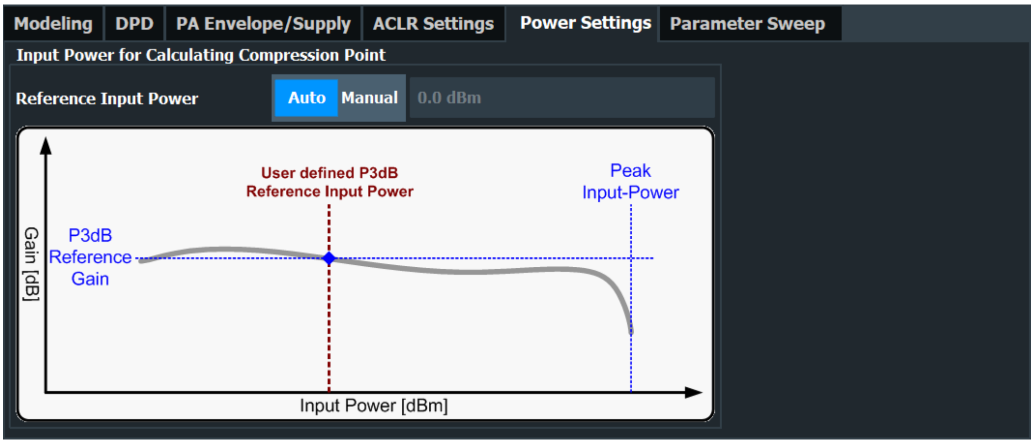
Parameter A / B
Undocumented feature.

Remote command:
`CONFigure:PAE:PCONsumption[:PARameter]:A`
`CONFigure:PAE:PCONsumption[:PARameter]:B`

3.16 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 5.6.18, "Configuring power measurements", on page 305.

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).

In **automatic** mode, the FSW-K18 application tries to automatically find the constant gain (linear range) section using a dedicated algorithm.

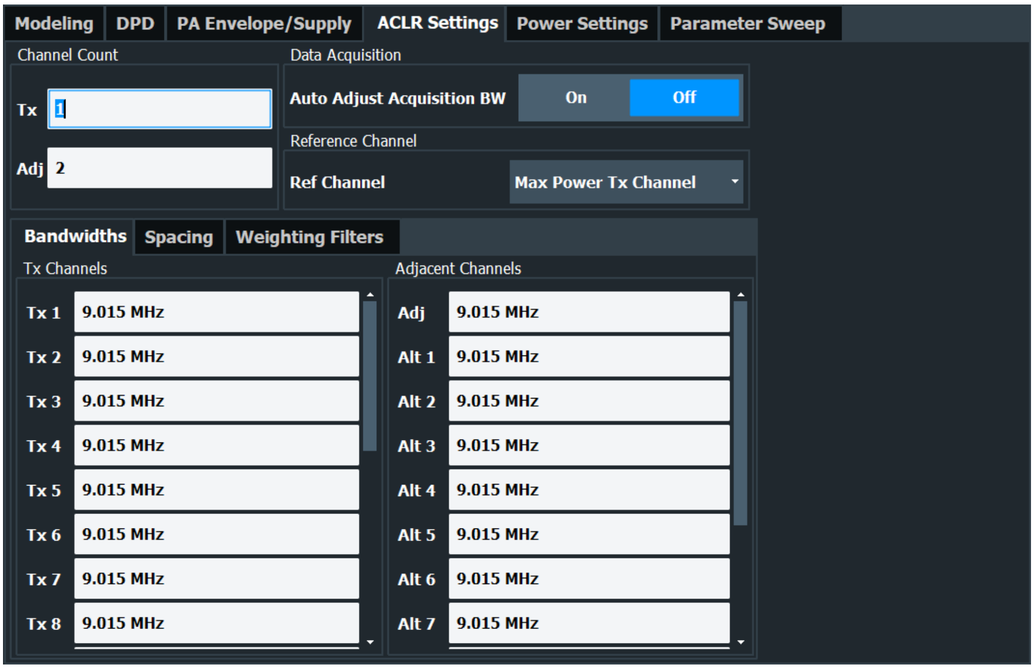
In the "Gain Compression" result display, the reference point is indicated by a red line.

Remote command:
Method: `CONFigure:POWer:RESult:P3DB[:STATe]` on page 305
Input power: `CONFigure:POWer:RESult:P3DB:REFerence` on page 305

3.17 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 5.6.17, "Configuring ACLR measurements", on page 299.

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 130).

Remote command:
Number of Tx channels:
`[SENSe:]POWer:ACHannel:TXCHannel:COUNt` on page 304
Number of Adjacent channels:
`[SENSe:]POWer:ACHannel:ACPairs` on page 300

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 FSW does not evaluate measurement results. Also make sure that the FSW 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 300

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 303
`[SENSe:]POWer:ACHannel:REFerence:TXCHannel:AUTO` on page 303

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 301
`[SENSe:]POWer:ACHannel:BANDwidth:ACHannel` on page 300
`[SENSe:]POWer:ACHannel:BANDwidth:ALTernate<ch>` on page 301

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 303
`[SENSe:]POWer:ACHannel:SPACing[:ACHannel]` on page 304
`[SENSe:]POWer:ACHannel:SPACing:ALTernate<ch>` on page 304

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 303
`[SENSe:]POWer:ACHannel:FILTer[:STATe]:ACHannel` on page 302
`[SENSe:]POWer:ACHannel:FILTer[:STATe]:ALTernate<ch>` on page 302
Alpha value:
`[SENSe:]POWer:ACHannel:FILTer:ALPHa:CHANnel<ch>` on page 302
`[SENSe:]POWer:ACHannel:FILTer:ALPHa:ACHannel` on page 301
`[SENSe:]POWer:ACHannel:FILTer:ALPHa:ALTernate<ch>` on page 302

3.18 Parameter sweeps

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 Table).

Example:

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

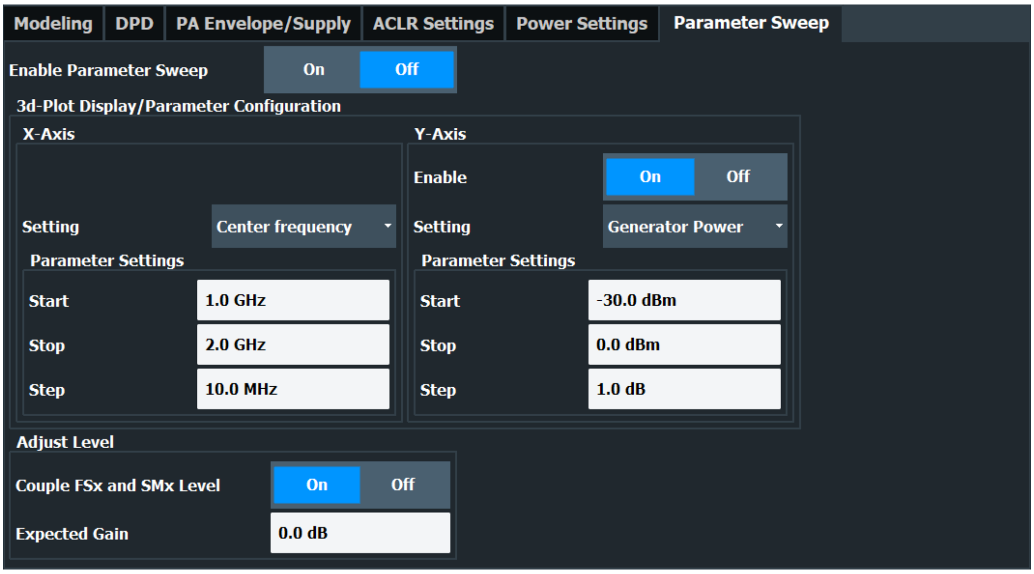
In that case, the FSW first performs a measurement on the first frequency for each generator output level in the defined range. When this measurement is done, the FSW continues to measure all power levels on the second frequency and so on.

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 3.4.7, "Controlling a signal generator", on page 79.



The remote commands required to configure the parameter sweep are described in Chapter 5.6.19, "Configuring parameter sweeps", on page 305.

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 FSW 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 306

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 FSW 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 FSW 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 FSW 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 5.6.19, "Configuring parameter sweeps", on page 305

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 FSW 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.

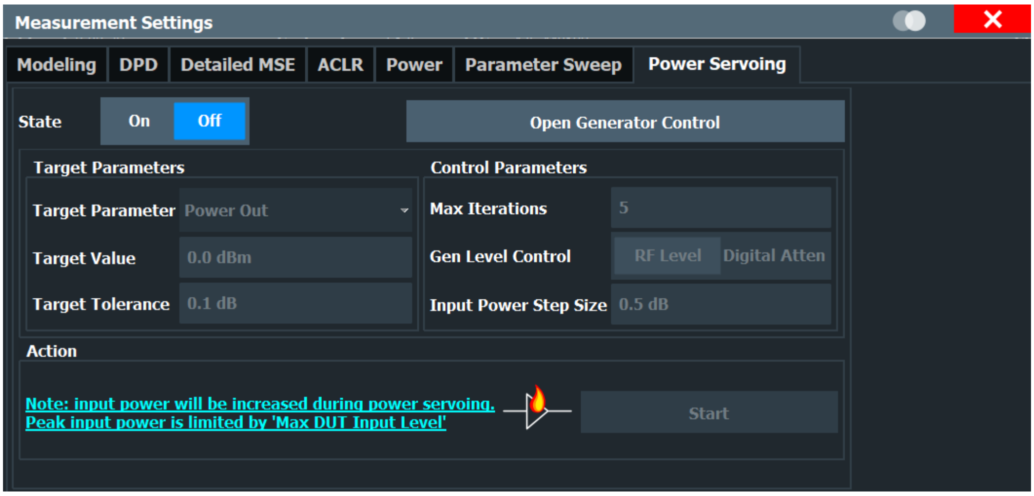
With a correct "Expected Gain" value, however, the application is able to attenuate the signal accordingly.

Remote command:
Synchronization state: `CONFigure:PSWeep:ADJust:LEVel[:STATe]` on page 306
Expected gain: `CONFigure:PSWeep:EXPected:GAIN` on page 306

3.19 Power servoing

Access: "Overview" > "Measurement" > "Power Servoing"

The power servoing measurement is an independent measurement mode that brings the output power of the DUT (amplifier) to a stable state (or level). This is done by automatically adjusting the input power or the digital attenuation until a stable state has been reached. When this state has been reached, you can proceed with measuring the actual amplifier performance.



The remote commands required to configure the power servoing are described in Chapter 5.6.20, "Configuring power servoing", on page 309.

Power Servoing sequence

To start a power servoing sequence, first enable the function using the "State" button. "Open Generator Control" provides a quick access to the generator setup dialog, e.g. to set the "Max DUT Input Level" to prevent damage to the DUT caused by a too high input level.

Now you can select a "Target Parameter" that is modified during the power servoing sequence until the desired "Target Value" is reached within the defined "Target Toler-ance".

The maximum number of repetitions performed during the sequence depends on the number of "Max Iterations" you have defined. Using "Gen Level Control", you can select if the FSW adjusts the input power or the digital attenuation until a stable state is reached. "Input Power Step Size" defines the maximum allowed input power change for each step for all "Target Parameters" except from "Power Out".

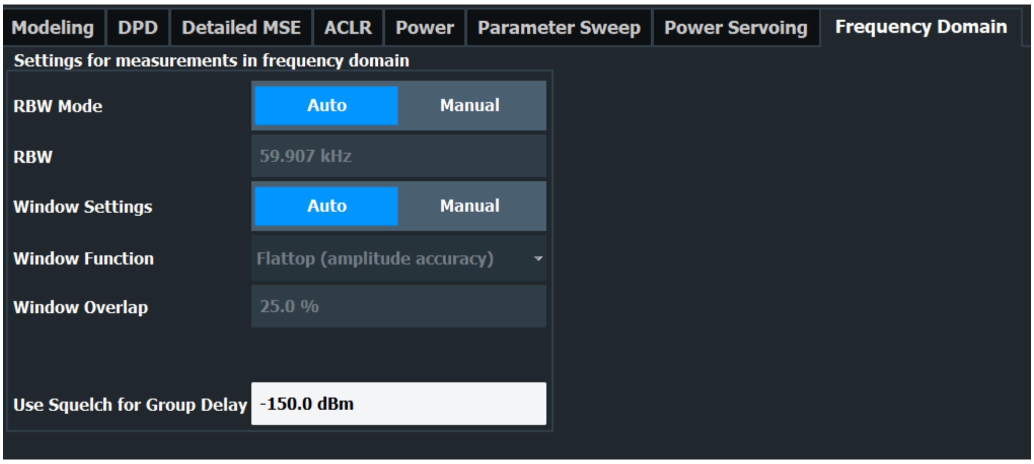
After all settings have been made, use "Start" to run the power servoing sequence.

Remote command:
State: `[SENSe:]PSERvoing:STATe` on page 310
Target Parameter: `[SENSe:]PSERvoing:TARGet:PARameter` on page 310
Target Value: `[SENSe:]PSERvoing:TARGet:VALue` on page 311
Target Tolerance: `[SENSe:]PSERvoing:TARGet:TOLerance` on page 311
Max Iterations: `[SENSe:]PSERvoing:MAX:ITERation` on page 311
Gen Level Control: `[SENSe:]PSERvoing[:GLC]` on page 311
Input Power Step Size: `[SENSe:]PSERvoing:INPut:STEP` on page 311
Start: `[SENSe:]PSERvoing:STARt` on page 312
Status: `FETCh:PSERvoing:OPERation:STATus?` on page 312

3.20 Frequency domain

Access: "Overview" > "Measurement" > "Frequency Domain"

You can configure how the results in the frequency domain are calculated, e.g. the FFT behavior to transform time values to frequency results.



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" allows defining an RBW value in "Manual" mode and determines an automatic RBW corresponding to a 2k FFT in "Auto" Mode.

When you select a 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 DSP configures the FFT automatically based on the RBW setting and window type.

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

The RBW mode selection determines the number of trace points. For more information, see Chapter 4.1.3, "Trace detector", on page 144.

Remote command:
`[SENSe<ip>:]BANDwidth[:RESolution]:AUTO` on page 255
`[SENSe<ip>:]BANDwidth[:RESolution]` on page 254

Window Settings

The window settings are set to automatic by default.

Select "Manual" to activate the "Window Function" and "Window Overlap" settings.

Remote command:
CONFigure:FDOMain:WSETtings:AUTO on page 313

Window Function
You can select one of several FFT window types with different characteristics.

The following window types are available:
- Flattop (default, high amplitude accuracy)
- Gauss (high dynamic range)
- Rectangular (no window)
- 5-Term (minimal sidelobes)
- Blackman-Harris

Remote command:
CONFigure:FDOMain:WFUNction on page 313

Window Overlap
Defines the part of a single FFT window that is re-calculated by the next FFT calculation when using multiple FFT windows.

Values from 0 % to 99.9 % in 1 % steps are supported. The default value is 25 %.

Remote command:
CONFigure:FDOMain:WOVerlap on page 314

Use Squelch for Group Delay
For group delay results (requires R&S FSW-K18F), you can define a level threshold below which the group delay is set to 0. If the group delay does not exceed the threshold, it is ignored altogether. Squelching prevents strong fluctuation in phase vs. group delay diagrams. If enabled, squelching is also considered for "Channel Response Magnitude" and "Channel Response Phase" results.

Values from -200 dBm to +200 dBm are supported.

Remote command:
CONFigure:FDOMain:SQUelch on page 313

3.21 Adjusting settings automatically

Access: [AUTO SET]

Some settings can be adjusted by the FSW automatically according to the current measurement settings. When you select an auto adjust function, a measurement is performed to determine the optimal settings.

For more available automatic settings, refer to the FSW user manual.

Before using the automatic settings, it is recommended to close all result windows and disabling all results that are not required.

Setting the Reference Level Automatically (Auto Level)
Automatically determines a reference level which ensures that no overload occurs at the FSW for the current input data. At the same time, the internal attenuators and the preamplifier (for analog baseband input: the full-scale level) are adjusted. As a result, the signal-to-noise ratio is optimized, while signal compression and clipping are minimized.

To determine the required reference level, a level measurement is performed on the FSW.

If necessary, you can optimize the reference level further. Decrease the attenuation level manually to the lowest possible value before an overload occurs, then decrease the reference level in the same way.

Remote command:
[SENSe:]ADJust:LEVel on page 314

Optimize the evaluation range settings for robustness (Auto Robust)
Maximizes the robustness of the measurement.

Pressing the "Auto Robust" button triggers the following settings:
- "Synchronization" > "Use Full Ref Signal": "On"
- "Evaluation Range" > "Use Full Ref Signal": "On"
- "Synchronization" > "2nd Stage Synchronization": "On"
- "I/Q Averaging": Turns "I/Q Avg" on, if not already switched on.
  – State: "On"
  – Statistics mode: "I/Q Avg"
  – Count: "10".
- "Synchronization" > "Synchronization Mode": "I/Q Phase Difference"
- "Synchronization Confidence": set to default.
- Switches all "Estimation" and "Compensation" settings to "On".
- "Direct DPD" > "Gain Expansion": "3.0 dB"

Remote command:
[SENSe:]AUTO:ROBust on page 314

Optimize the evaluation range settings for speed (Auto Fast)
Maximizes the speed of the measurement.

"Auto Fast" uses an algorithm to extract a representative segment of 30,000 samples from the reference signal. This segment is used for all measurements to improve the measurement speed.

Note: "Auto Fast" is only available if reference signal is loaded, otherwise it is grayed out.

Pressing the "Auto Fast" button triggers the following settings:
- "Synchronization" > "Use Full Ref Signal": "Off"
- "Estimation Start": uses the time stamps as determined by the algorithm.
- "Estimation Stop": uses the time stamps as determined by the algorithm.
- "Evaluation Range" > "Use Full Ref Signal": "Off"
- "Eval Start": uses the time stamps as determined by the algorithm.
- "Eval Stop": uses the time stamps as determined by the algorithm.
- "Synchronization" > "2nd Stage Synchronization": "Off"
- "I/Q Averaging": "Off" (regardless of statistics mode)

- Switches all "Estimation" settings, except "Frequency Error" and "Sample Rate Error", to "Off".
- "Modelling": "Off"

Remote command:
`[SENSe:]AUTO:FAST` on page 314

4 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 FSW user manual.

4.1 Traces

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

4.1.1 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 2, "Measurements and result displays", on page 12.

Trace Mode

Defines the update mode for subsequent traces.

Trace modes (except for "Clear Write", "View" and "Blank") are only available if detector is set to "Positive Peak", "Negative Peak" or "Average" and trace statistics are enabled.

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 FSW 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 FSW 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 315

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 319

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 318

Predefined Trace Settings - Quick Config

Commonly required trace settings have been predefined and can be applied very quickly by selecting the appropriate button.

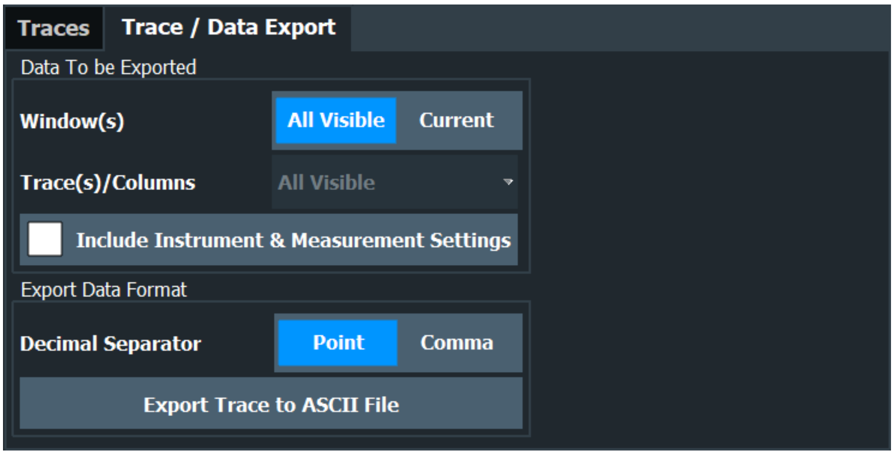
Function	Trace Settings	
Preset All Traces	Trace 1:	Clear Write
		Blank
Set Trace Mode Max | Avg | Min	Trace 1:	Max Hold
	Trace 2:	Average
	Trace 3:	Min Hold
		Blank
Set Trace Mode Max | ClrWrite | Min	Trace 1:	Max Hold
	Trace 2:	Clear Write
	Trace 3:	Min Hold
		Blank

Remote command:
DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:PRESet on page 318

4.1.2 Trace export

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

The functionality to export traces is similar to the Spectrum application. When you export a trace, the FSW 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 5.7.1, "Configuring traces", on page 315.

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 317

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 316

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 316

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.

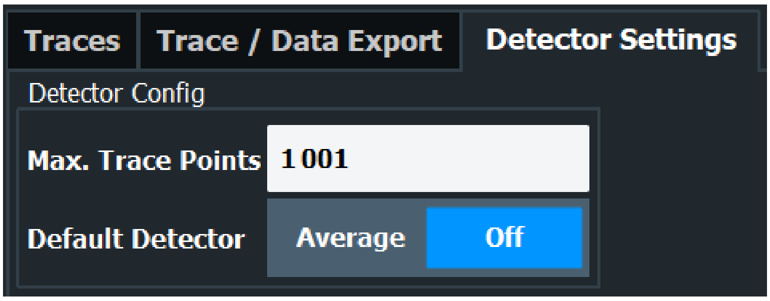
Note: Secure user mode.
In secure user mode, settings that are stored on the instrument are stored to volatile memory, which is restricted to 256 MB. Thus, a "memory limit reached" error can occur although the hard disk indicates that storage space is still available.

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 FSW user manual.

Remote command:
`MMEMory:STORe<n>:TRACe` on page 317

4.1.3 Trace detector



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

The number of trace points is determined automatically by the RWB selection:

- "Auto": the number of trace points will correspond to the "Max. Trace Points".
- "Manual": the number of trace points are equal or less than the number selected for "Max. Trace Points".

Remote command:
`[SENSe:]DETector<t>:TRACe[:POINt]` on page 318

Default Detector
Selects the default detector for FSW-K18 result displays.

Note that changing the default detector to "Average" changes all traces with "Detector Off" to "Detecor Average". Setting "Off" will change all traces with "Detector Average" to "Detector Off"

Remote command:
[SENSe:]DETector<t>:DEFault[:FUNCtion] on page 317

4.2 Markers

The amplifier application provides four markers in most result displays.

4.2.1 General marker settings

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.

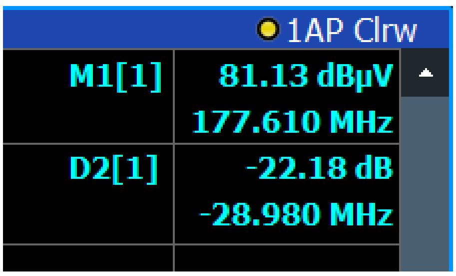
"Off" No separate marker table is displayed.
If Marker Info is active, the marker information is displayed within the diagram area.

"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 321

Marker Info

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



Remote command:
DISPlay[:WINDow<n>]:MINFo[:STATe] on page 321

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).

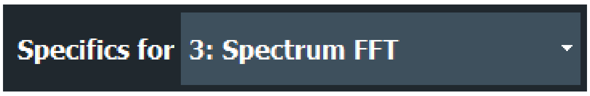
Remote command:
CALCulate<n>:MARKer<m>:LINK on page 320

4.2.2 Individual marker settings

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 326
`CALCulate<n>:DELTamarker<m>[:STATe]` on page 323

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 326
`CALCulate<n>:DELTamarker<m>:X` on page 324

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 326
`CALCulate<n>:DELTamarker<m>[:STATe]` on page 323

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 323

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).

For linked delta markers, the x-value of the delta marker is 0 Hz by default. To create a delta marker in a fixed distance to another marker, define the distance as the x-value for the linked delta marker.

Remote command:
`CALCulate<n>:MARKer<ms>:LINK:TO:MARKer<md>` on page 325
`CALCulate<n>:DELTamarker<ms>:LINK:TO:MARKer<md>` on page 323
`CALCulate<n>:DELTamarker<m>:LINK` on page 322

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 326

All Markers Off
Deactivates all markers in one step.

Remote command:
`CALCulate<n>:MARKer<m>:AOFF` on page 325

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 321

4.2.3 Marker positioning

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 330
CALCulate<n>:DELTamarker<m>:MAXimum[:PEAK] on page 328

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 330
CALCulate<n>:MARKer<m>:MAXimum:RIGHt on page 330
CALCulate<n>:MARKer<m>:MAXimum:LEFT on page 329
CALCulate<n>:DELTamarker<m>:MAXimum:NEXT on page 328
CALCulate<n>:DELTamarker<m>:MAXimum:RIGHt on page 328
CALCulate<n>:DELTamarker<m>:MAXimum:LEFT on page 327

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 331
CALCulate<n>:DELTamarker<m>:MINimum[:PEAK] on page 329

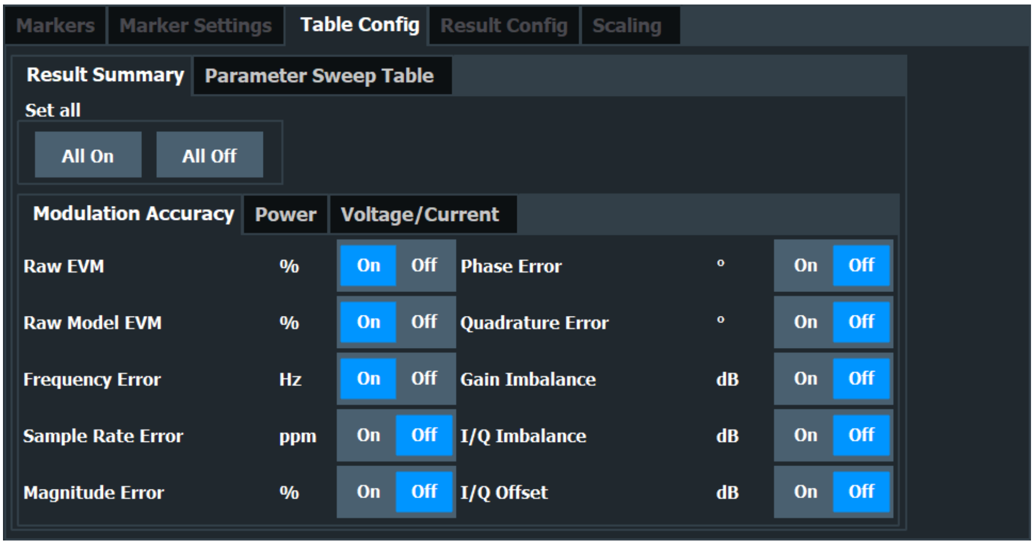
Search Next Minimum
Sets the selected marker/delta marker to the next (higher) minimum of the selected trace. If no marker is active, marker 1 is activated.

Remote command:
CALCulate<n>:MARKer<m>:MINimum:NEXT on page 331
CALCulate<n>:MARKer<m>:MINimum:LEFT on page 330
CALCulate<n>:MARKer<m>:MINimum:RIGHt on page 331
CALCulate<n>:DELTamarker<m>:MINimum:NEXT on page 329
CALCulate<n>:DELTamarker<m>:MINimum:LEFT on page 328
CALCulate<n>:DELTamarker<m>:MINimum:RIGHt on page 329

4.3 Numerical result tables

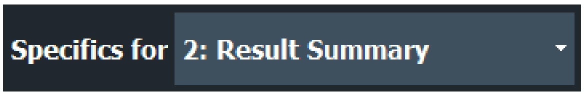
Access: "Overview" > "Result Config" > "Table Config"

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 different tabs:

- One for modulation accuracy results.
- One for power-related results.
- One for voltage and current related results. The results in this tab are available after you have activated baseband measurements.

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 332

Individual parameter sweep items: `DISPlay[:WINDow<n>]:PTABle:ITEM` on page 331

All modulation accuracy items: `DISPlay[:WINDow<n>]:TABLe:ITEM:MACCuracy:ALL` on page 334

All power items: `DISPlay[:WINDow<n>]:TABLe:ITEM:POWer:ALL` on page 334

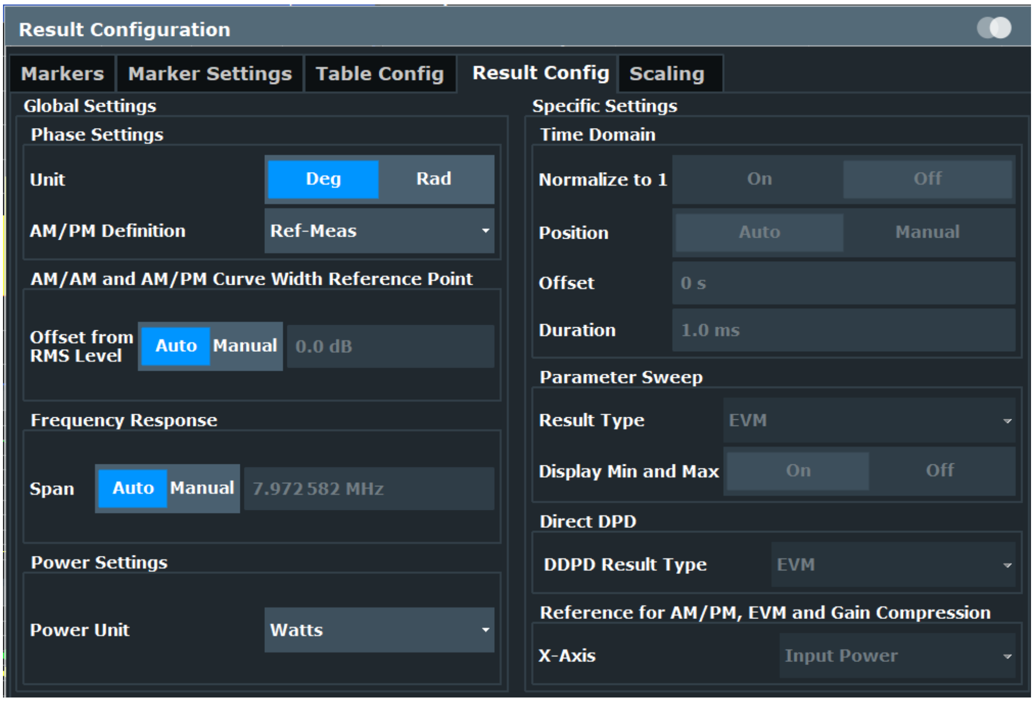
All voltage / current items: `DISPlay[:WINDow<n>]:TABLe:ITEM:VCURrent:ALL` on page 334

All parameter sweep items: `DISPlay[:WINDow<n>]:PTABle:ITEM:ALL` on page 332

4.4 Result display settings

Access: "Overview" > "Result Config"

The application allows you to configure the information displayed in various graphical result displays. The global settings apply to all relevant result displays, while the specific settings apply only to the selected result display ("Specifics for").



Phase Settings

Phase settings apply to the "AM/PM" results and the "Phase deviation vs time" result.

Unit ← Phase Settings
Determines whether phase information is provided in degrees or radians.

Remote command:
`CALCulate<n>:UNIT:ANGLe` on page 337

AM/PM Definition ← Phase Settings
Determines the way the "AM/PM" results are calculated.

"Ref - Meas" (Default:) The measured values are subtracted from the reference values.

"Meas - Ref" The reference values are subtracted from the measured values.

Remote command:
`CALCulate<n>:AMPM:DEFinition` on page 336

AM/AM and AM/PM Curve Width Reference Point
Determines the reference point for the curve width as an offset from the RMS level. By default, the input power is automatically set to the RMS level. Positive values describe a value above RMS power and negative values below RMS power.

Remote command:
`CONFigure:AMPM:CWIDth:REFerence:AUTO` on page 341
`CONFigure:AMPM:CWIDth:REFerence` on page 341

Frequency Response
Selects the span that the frequency response is applied to for FSW-K18F result displays.
- "Auto": Sets the span to the calculated OBW of the reference file.
- "Manual": Define the span manually.

Remote command:
`CONFigure:FRSPan` on page 338
`CONFigure:FRSPan:AUTO` on page 338

Power Unit
Switches the unit for power results from dBm (default) to Watts.

The following windows are then adjusted accordingly.
- "Result Summary": Adjusts all affected units in the result summary accordingly.
- "Magnitude Capture" and "Spectrum": Adjusts all axes for graphical results accordingly.
- "AM/AM": X-axis in dBm
- "AM/PM": X-axis in dBm
- "Gain Compression": X-axis in dBm
- "EVM (Input Power)": X-axis in dBm

Remote command:
`CONFigure:POWer:UNIT` on page 338

Time domain result display
The "Time Domain" settings configure the information displayed in the time domain result display, e.g.the displayed time range.

Normalize to 1 ← Time domain result display
Enables or disables normalization of the time domain trace to 1.

Remote command:
`DISPlay[:WINDow<n>]:TDOMain:Y[:SCALe]:NORMalise[:STATe]`
on page 341

Position ← Time domain result display
Defines whether the position of the time domain range is determined automatically or manually, by the Offset and Duration.

Remote command:
`DISPlay[:WINDow<n>]:TDOMain:X[:SCALe]:MODE` on page 340

Offset ← Time domain result display
Defines an offset to the time of the first recorded sample or the first sample of the synchronized data. The x-axis of the time domain result display starts at this value.

This setting is only available for manual positioning (see "Position" on page 153).

Remote command:
`DISPlay[:WINDow<n>]:TDOMain:X[:SCALe]:OFFSet` on page 340

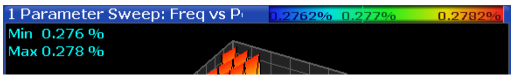
Duration ← Time domain result display
Defines the length of the x-axis in time domain results.

This setting is only available for manual positioning (see "Position" on page 153).

Remote command:
`DISPlay[:WINDow<n>]:TDOMain:X[:SCALe]:DURation` on page 339

Parameter sweep diagram
You can select one of several result types to evaluate in the parameter sweep diagram. If you open more than one parameter sweep window, you can select a different result for each window.

By default, the application indicates the highest and lowest values that have been measured in the diagram.



Optionally, you can disable the "Display Min and Max" feature.

The following result types are evaluated in the parameter sweep.
- "ACLR Adj 1 Lower"
- "ACLR Adj 1 Upper"
- "ACLR Adj 1 Balanced"
- "ACLR Alt 1 Balanced"
- "ACLR Alt 2 Balanced"
- "AM/AM Curve Width"
- "AM/PM Curve Width"
- "Balanced ACLR Magnitude"
- Compression Point "P (1 dB / 2 dB / 3 dB)"
- "Crest Factor Out"
- "Current OBW"

- "EVM"
- "Gain"
- "Power Out"
- "RMS Power"
- "Voltage (V_cc)"
- "Current (I_cc)"
- "Power (V_cc * I_cc)"

For details on the parameters, see Chapter 2.2, "Amplifier parameters", on page 33.

Remote command:
`CONFigure:PSWeep:Z<n>:RESult` on page 338

Direct DPD Result Type
Selects the result type for direct DPD measurements.

For a description of the supported result types, see "DDPD Results (FSW-K18D)" on page 18.

Remote command:
`CONFigure:DDPD:WINDow<n>:RESult` on page 287

Reference for AM/PM, EVM and Gain Compression
For the following power result displays, you can select the information that is used as a reference for the x-axis.

- AM/PM
- EVM vs Power
- Gain Compression
- Vcc Power
- Icc Power

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".

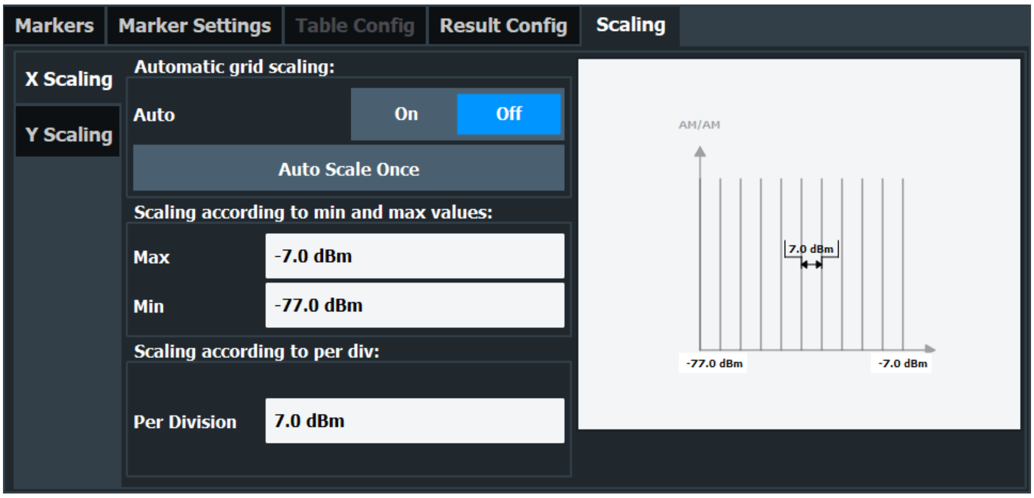
To analyze the signal against the output power, select "Output Power".

Remote command:
`CALCulate<n>:PREFerence:X` on page 337

4.5 X-axis scaling

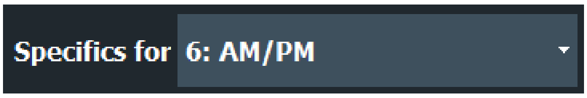
Access: "Overview" > "Result Config" > "Scaling" > "X Scaling"

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 342

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.

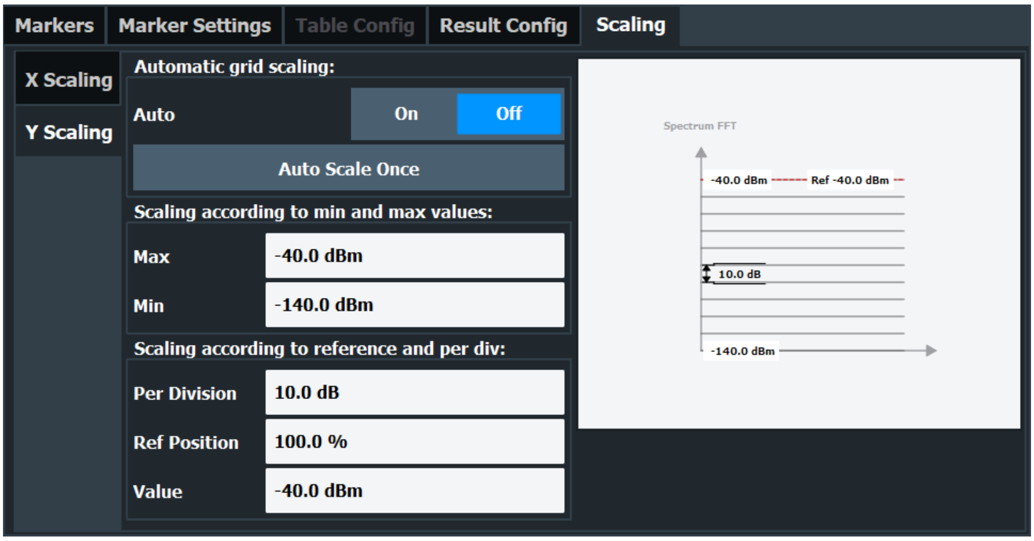
- Scaling according to the distance between two grid lines
  The scale is defined by the value range within two grid lines in the diagram (➙ per division). The distance between grid lines refers to diagrams that are split into 10 divisions.

Remote command:
Minimum: `DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:X[:SCALe]:MINimum` on page 343
Maximum: `DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:X[:SCALe]:MAXimum` on page 342
Distance: `DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:X[:SCALe]:PDIVision` on page 343

4.6 Y-axis scaling

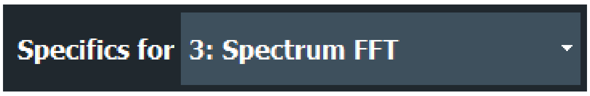
Access: "Overview" > "Result Config" > "Scaling" > "Y Scaling"

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 344

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 345
Maximum: DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:MINimum on page 345
Reference value: DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:RVALue on page 347
Position: DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:RPOSition on page 346
Distance: DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:PDIVision on page 346

5 Remote control commands for amplifier measurements

The following remote control commands are required to configure and perform amplifier measurements in a remote environment. The FSW 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 FSW 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).

SCPI Recorder - automating tasks with remote command scripts

The Amplifier measurement application also supports the SCPI Recorder functionality.

Using the SCPI Recorder functions, you can create a SCPI script directly on the instrument and then export the script for use on the controller. You can also edit or write a script manually, using a suitable editor on the controller. For manual creation, the instrument supports you by showing the corresponding command syntax for the current setting value.

For details see the "Network and Remote Operation" chapter in the FSW User Manual.

5.1 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, usually, 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, they 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 FSW.

Remote command examples

Note that some remote command examples mentioned in this general introduction are possibly not supported by this particular application.

5.1.1 Conventions used in descriptions

The following conventions are used in the remote command descriptions:

- *Command usage*
  If not specified otherwise, commands can be used both for setting and for querying parameters.
  If a command can be used for setting or querying only, or if it initiates an event, the usage is stated explicitly.
- *Parameter usage*
  If not specified otherwise, a parameter can be used to set a value, and it is the result of a query.
  Parameters required only for setting are indicated as "Setting parameters".
  Parameters required only to refine a query are indicated as "Query parameters".
  Parameters that are only returned as the result of a query are indicated as "Return values".
- *Conformity*
  Commands that are taken from the SCPI standard are indicated as "SCPI confirmed". All commands used by the FSW 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.

5.1.2 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 uppercase 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`.

5.1.3 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 do not 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.

5.1.4 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.

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.

5.1.5 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.

5.1.6 SCPI parameters

Many commands feature one or more parameters.

If a command supports more than one parameter, they are separated by a comma.

Example:

LAYout:ADD:WINDow Spectrum,LEFT,MTABle

Parameters can have different forms of values.

5.1.6.1 Numeric values

Numeric values can be entered in any form, i.e. with sign, decimal point or exponent. For 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. for 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. Sometimes, you can customize the step size with a corresponding command.

Querying numeric values

When you query numeric values, the system returns a number. For physical quantities, it applies the basic unit (e.g. Hz for 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`

Sometimes, numeric values are 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 if errors occur.

5.1.6.2 Boolean

Boolean parameters represent two states. The "on" state (logically true) is represented by "`ON`" or the 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`

5.1.6.3 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 5.1.2, "Long and short form", on page 160.

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`

5.1.6.4 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'`

5.1.6.5 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. 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.

5.2 Common suffixes

In the Amplifier measurement application, the following common suffixes are used in remote commands:

Table 5-1: 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**)

Suffix	Value range	Description
<t>	1..6	Trace
	1 to 8	Limit line

5.3 Selecting the application

INSTrument:CREate:DUPLicate

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>

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 165.

<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 165).
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>

Adds a 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 165.

<ChannelName> String containing the name of the channel.
Note that you cannot assign an existing channel name to a new channel. If you do, an error occurs.

Example: `INST:CRE SAN, 'Spectrum 2'`
Adds a spectrum display named "Spectrum 2".

INSTrument:DELete <ChannelName>

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 to delete it.

Example: `INST:DEL 'IQAnalyzer4'`
Deletes the channel with the name 'IQAnalyzer4'.

Usage: Setting only

INSTrument:LIST?

Queries all active channels. The query is useful to obtain the names of the existing channels, which are required 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 5-2: 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 FSW-K84)	BDO	1xEV-DO BTS
1xEV-DO MS (R&S FSW-K85)	MDO	1xEV-DO MS
3GPP FDD BTS (R&S FSW-K72)	BWCD	3G FDD BTS
3GPP FDD UE (R&S FSW-K73)	MWCD	3G FDD UE
802.11ad (R&S FSW-K95)	WIGIG	802.11ad
802.11ay (R&S FSW-K97)	EDMG	802.11ay EDMG
Amplifier Measurements (R&S FSW-K18)	AMPLifier	Amplifier
AM/FM/PM Modulation Analysis (R&S FSW-K7)	ADEM	Analog Demod
Avionics (R&S FSW-K15)	AVIonics	Avionics
Bluetooth (R&S FSW-K8)	BTO	Bluetooth
cdma2000 BTS (R&S FSW-K82)	BC2K	CDMA2000 BTS
cdma2000 MS (R&S FSW-K83)	MC2K	CDMA2000 MS
DOCSIS 3.1 (R&S FSW-K192/193)	DOCSis	DOCSIS 3.1
Fast Spur Search (R&S FSW-K50)	SPUR	Spurious
GSM (R&S FSW-K10)	GSM	GSM
HRP UWB (R&S FSW-K149)	UWB	HRP UWB
I/Q Analyzer	IQ	IQ Analyzer
LTE (R&S FSW-K10x)	LTE	LTE
Multi-Carrier "Group Delay" (R&S FSW-K17)	MCGD	MC "Group Delay"
NB-IoT (R&S FSW-K106)	NIOT	NB-IoT
Noise (R&S FSW-K30)	NOISE	Noise
*) 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*)
5G NR (R&S FSW-K144)	NR5G	5G NR
OFDM VSA (R&S FSW-K96)	OFDMVSA	OFDM VSA
OneWeb (R&S FSW-K201)	OWEB	OneWeb
Phase Noise (R&S FSW-K40)	PNOISE	Phase Noise
Pulse (R&S FSW-K6)	PULSE	Pulse
"Real-Time Spectrum"	RTIM	"Real-Time Spectrum"
TD-SCDMA BTS (R&S FSW-K76)	BTDS	TD-SCDMA BTS
TD-SCDMA UE (R&S FSW-K77)	MTDS	TD-SCDMA UE
Transient Analysis (R&S FSW-K60)	TA	Transient Analysis
Verizon 5GTF Measurement Application (V5GTF, R&S FSW-K118)	V5GT	V5GT
VSA (R&S FSW-K70)	DDEM	VSA
WLAN (R&S FSW-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>

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. If you do, an error occurs.
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>

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 165

Parameters:

<ChannelType> Channel type of the new channel.
For a list of available channel types see INSTrument:LIST? on page 165.

<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]

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 46

5.4 Configuring the screen layout

DISPlay:FORMat <Format>

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>

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 173).

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>

Adds a window to the display in the active channel.

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 "Magnitude Capture" on page 22
See "Spectrum FFT" on page 29

Table 5-3: <WindowType> parameter values for Amplifier Measurement application

Parameter value	Window type
ACP	"Adjacent Channel Power" (Table)
AMAM	"AM/AM"
AMEVm	"EVM" vs Power
AMPM	"Phase Deviation vs Input Power"("AM/PM")
DDPD	DDPD Results
GCOMpression	"Gain Compression"
GDEL	"Group Delay"
GDVTime	"Gain Deviation vs Time"
IMAGnitude	"Magnitude Capture" I
ICCPower	Icc vs Power
ISPectrum	"Spectrum FFT" I
MDPD	"Memory DPD Coefficients"
MRES	Channel Response Magnitude
MTABle	"Marker Table"
PAEI	PAE Input Power
PAEO	PAE Output Power
PAETime	PAE Time
PRES	Channel Response Phase

Parameter value	Window type
PSWeep	"Parameter Sweep" (Diagram)
PTABle	"Parameter Sweep" (Table)
PDVTime	"Phase Deviation vs Time"
PVTime	Power vs. Time (I x Q)
QMAGnitude	"Magnitude Capture" Q
QSPectrum	"Spectrum FFT" Q
REVM	Raw "EVM"
RFMagnitude	"Magnitude Capture" RF
RFSPectrum	"Spectrum FFT"
RTABle	"Result Summary" (Table)
SEVM	Spectrum "EVM"
STAB	"Statistics Table"
TDOMain	"Time Domain"
VCCPower	Vcc vs Power
VICC	Vcc vs. Icc

LAYout:CATalog[:WINDow]?

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>

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:IDEN:WIND? '2'`
Queries the index of the result display named '2'.
Response:
`2`

Usage: Query only

LAYout:REMove[:WINDow] <WindowName>

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>

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 169 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>

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 169 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 does not work, but does not return an error.

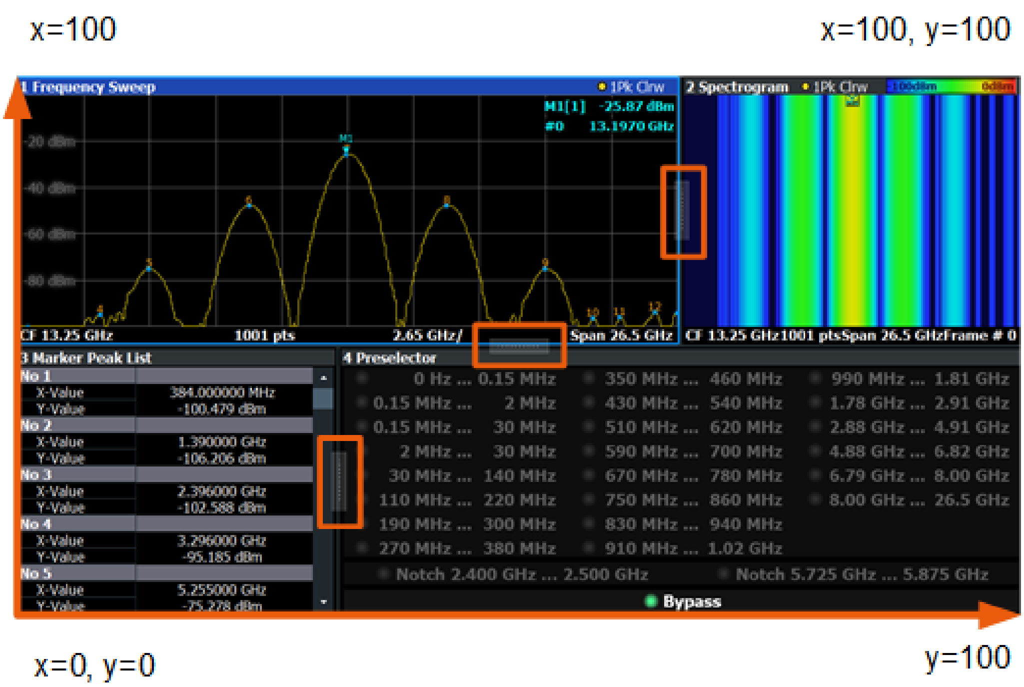


Figure 5-1: SmartGrid coordinates for remote control of the splitters

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 5-1.) 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>

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. Unlike `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.

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 169 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?

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

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>

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 169 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 169.

Suffix:
<n> Window

Example: LAY:WIND2:TYPE?

Usage: Query only

5.5 Performing amplifier measurements

5.5.1 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 FSW User Manual.

INITiate<n>:CONMeas

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

Usage: Asynchronous command

Manual operation: See "Continue Single Sweep" on page 48

INITiate<n>:CONTinuous <State>

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.

For details on synchronization see Remote control via SCPI.

Suffix:
<n> irrelevant

Parameters:
<State> ON | OFF | 0 | 1

ON | 1
Continuous measurement

OFF | 0
Single measurement

*RST: 1 (some applications can differ)

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 47

INITiate<n>[:IMMediate]

Starts a (single) new measurement.

You can synchronize to the end of the measurement with *OPC, *OPC? or *WAI.

For details on synchronization see Remote control via SCPI.

Suffix:
<n> irrelevant

Usage: Asynchronous command

Manual operation: See "Single Sweep / Run Single" on page 47

INITiate:SEQuencer:ABORt

Stops the currently active sequence of measurements.

You can start a new sequence any time using `INITiate:SEQuencer:IMMediate` on page 178.

Usage: Event

INITiate:SEQuencer:IMMediate

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 179).

Example:
`SYST:SEQ ON`
Activates the Sequencer.
`INIT:SEQ:MODE SING`
Sets single sequence mode so each active measurement is 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: To synchronize to the end of a measurement sequence using `*OPC`, `*OPC?` or `*WAI`, use `SINGle` Sequencer 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>

Turns the Sequencer on and off. The Sequencer must be active before any other Sequencer commands (`INIT:SEQ...`) are executed, otherwise an error occurs.

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 is performed once.
`INIT:SEQ:IMM`
Starts the sequential measurements.
`SYST:SEQ OFF`

[SENSe:]ADJust:TIME:TRIGger

Calculates the time trigger.

Example: `[SENSe]:ADJust:TIMe:TRIGger`

Usage: Event

5.6 Configuring amplifier measurements

5.6.1 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 51

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 51

CONFigure:REFSignal:CGW:READ

This command transfers a reference signal designed on a signal generator into the R&S FSW-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 51

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 52

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 52

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 52

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 52

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 52

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 55

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 "Target Crest Factor" on page 56

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 56

CONFigure:REFSignal:GOS:FPATh <FilePath>

Defines the name and path of the user-defined reference waveform file when loaded to the analyzer.

Parameters:

<FilePath> String containing the path and name of the file.

Example: //Generator control disabled
//Define waveform file name and path on the analyzer
`CONF:REFS:GOS:FPATh 'C:\RefSignal.wv'`
//Create waveform file on generator and load it to analyzer in specified path
`CONF:REFS:GOS:WRIT`
Select the waveform file containing the reference signal
`CONF:REFS:CWF:FPAT 'C:\RefSignal.wv'`
//Load waveform file to analyzer in specified path
`CONF:REFS:CWF:WRITE;*WAI`

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 FSW-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 FSW-K18" on page 54

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 57

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 56

CONFigure:REFSignal:GOS:PATH <FileName>

Defines the path to load user-defined reference waveform files to. If you do not specify a path, the file is loaded to `C:\R_S\INSTR\USER\K18\ReferenceFiles`.

Parameters:

<FileName> String containing the path of the file.

Example:
//Enable generator control
`CONF:GEN:CONT ON`
//Define waveform file name
`CONFigure:REFSignal:GOS:WNAMe 'RefFile'`
//Define storage location for waveform file on analyzer after loading
`CONF:REFS:GOS:PATH 'c:\user\'`
//Create waveform file on generator and load it to analyzer in specified path
`CONFigure:REFSignal:GOS:WRIT`

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 56

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 56

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 56

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 FSW-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 FSW-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 FSW-K18" on page 54

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 50

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 49

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 49

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 49

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 49

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 49

CONFigure:CFReduction[:STATe] <State>

Enables the crest factor reduction calculation.

Parameters:
<State>

Example: `CONFigure:CFR ON`

Manual operation: See "Crest Factor Reduction State" on page 58

CONFigure:CFReduction[:STATe]:LEDState?

Reads the LED status of the crest factor reduction calculation.

Return values:
<State> GREY | RED | GREen

Example: `CONFigure:CFReduction:STATe:LEDState`

Usage: Query only

Manual operation: See "Crest Factor Reduction State" on page 58

CONFigure:CFReduction:SBANdwidth <Time>

Sets and queries the signal bandwidth.

Parameters:
<Time> <numeric value>
Default unit: Hz

Example: `CONF:CFR:SBAN 10MHz`

Manual operation: See "Signal Bandwidth" on page 59

CONFigure:CFReduction:SBANdwidth:AUTO <State>

Sets and queries the signal bandwidth mode.

Parameters:
<State>

Example: `CONFigure:CFReduction:SBANdwidth:AUTO ON`

Manual operation: See "Signal Bandwidth" on page 59

CONFigure:CFReduction:SBANdwidth:LEDState?

Reads the LED status of the signal bandwith.

Return values:
<State> GREY | RED | GREEn

Example: `CONFigure:CFReduction:SBANdwidth:LEDState`

Usage: Query only

Manual operation: See "Signal Bandwidth" on page 59

CONFigure:CFReduction:RSORignal <State>

Switches the EVM reference signal.

Parameters:
<State>

Example: `CONFigure:CFReduction:RSORignal ON`

Manual operation: See "EVM Ref. Signal" on page 59

CONFigure:CFReduction:ITERations <Iterations>

Sets and queries the crest factor reduction maximum iterations.

Parameters:
<Iterations> <numeric value>

Example: `CONFigure:CFReduction:ITERations 2`

Manual operation: See "Max Iterations" on page 59

CONFigure:CFReduction:ITERations:LEDState?

Reads the LED status of the crest factor reduction maximum iterations.

Return values:
<State> GREY | RED | GREEn

Example: `CONFigure:CFReduction:ITERations:LEDState`

Usage: Query only

Manual operation: See "Max Iterations" on page 59

CONFigure:CFReduction:FILTer <FilterMode>

Selects simple or enhanced filter mode for crest factor reduction.

Parameters:

<FilterMode>	SIMPle | ENHanced
Example:	`CONFigure:CFReduction:FILTer ENH`
Manual operation:	See "Filter Mode" on page 59

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
Manual operation:	See "Filter Mode" on page 59

CONFigure:CFReduction:CSPacing <Time>

Sets and queries the crest factor reduction channel spacing.

Parameters:

<Time>	<numeric value> Default unit: Hz
Example:	`CONF:CFR:CSP 10MHz`
Manual operation:	See "Channel Spacing" on page 60

CONFigure:CFReduction:CSPacing:AUTO <State>

Sets and queries the crest factor reduction channel spacing mode.

Parameters:

<State>	
Example:	`CONFigure:CFReduction:CSPacing:AUTO ON`
Manual operation:	See "Channel Spacing" on page 60

CONFigure:CFReduction:CSPacing:LEDState?

Reads the LED status of the crest factor reduction channel spacing.

Return values:

<State>	GREY | RED | GREen
Example:	`CONFigure:CFReduction:CSPacing:LEDState`
Usage:	Query only
Manual operation:	See "Channel Spacing" on page 60

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`

Manual operation: See "Crest Factor Delta" on page 59

CONFigure:CFReduction:CFDelta:LEDState?

Reads the LED status of the crest factor delta.

Return values:

<State> GREY | RED | GREen

Example: `CONFigure:CFReduction:CFDelta:LEDState`

Usage: Query only

Manual operation: See "Crest Factor Delta" on page 59

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

Manual operation: See "Current Crest Factor" on page 59

CONFigure:CFReduction:APPLy

Applies crest factor reduction on the connected signal generator.

Only available for backward compatibility, use `CONFigure:CFReduction:READ` on page 194 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 194 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

Manual operation: See "Read CFR from Generator, Load" on page 60

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

Manual operation: See "Read CFR from Generator, Load" on page 60

CONFigure:CFReduction:MFORder <MaximumFilterOrder>

Sets and queries the maximum filter order for crest factor reduction.

Parameters:
<MaximumFilterOrder> numeric value

Example: `CONF:CFR:MFOR 100`

Manual operation: See "Maximum Filter Order" on page 60

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

Manual operation: See "Maximum Filter Order" on page 60

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`

Manual operation: See "Passband Frequency" on page 60

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

Manual operation: See "Passband Frequency" on page 60

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`

Manual operation: See "Stopband Frequency" on page 60

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

Manual operation: See "Stopband Frequency" on page 60

5.6.2 Selecting and configuring the input source

The following commands are required to configure data input and output.

5.6.2.1 RF input

CALibration:AIQ:HATiming[:STATe] <State>

Activates a mode with enhanced timing accuracy between analog baseband, RF and external trigger signals.

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

Example: `CAL:AIQ:HAT:STAT ON`

Manual operation: See "High Accuracy Timing Trigger - Baseband - RF" on page 65

INPut:CONNector <ConnType>

Determines which connector the input for the measurement is taken from.

If an external frontend is active, the connector is automatically set to `RF`.

Parameters:

<ConnType>

RF
RF input connector

AIQI
Analog Baseband I connector
This setting is only available if the "Analog Baseband" interface (FSW-B71) is installed and active for input. It is not available for the FSW67 or FSW85.
For more information on the "Analog Baseband" interface (FSW-B71), see the FSW I/Q Analyzer and I/Q Input User Manual.

RFPRobe
Active RF probe

*RST: RF

Example: `INP:CONN RF`
Selects input from the RF input connector.

Manual operation: See "Input Connector" on page 64

INPut:COUPling <CouplingType>

Selects the coupling type of the RF input.

If an external frontend is active, the coupling is automatically set to `AC`.

Parameters:

<CouplingType> AC | DC

AC
AC coupling

DC
DC coupling

*RST: AC

Example: `INP:COUP DC`

Manual operation: See "Input Coupling" on page 62

INPut:DPATh <DirectPath>

Enables or disables the use of the direct path for frequencies close to 0 Hz.

If an external frontend is active, the direct path is always used.

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`

Manual operation: See "Direct Path" on page 63

INPut:FILE:PATH <FileName>[, <AnalysisBW>]

Selects the I/Q data file to be used as input for further measurements.

The I/Q data file must be in one of the following supported formats:

- `.iq.tar`
- `.iqw`
- `.csv`
- `.mat`
- `.wv`
- `.aid`

Only a single data stream or channel can be used as input, even if multiple streams or channels are stored in the file.

For some file formats that do not provide the sample rate and measurement time or record length, you must define these parameters manually. Otherwise the traces are not visible in the result displays.

Parameters:

<FileName> String containing the path and name of the source file.
The file type is determined by the file extension. If no file extension is provided, the file type is assumed to be `.iq.tar`.
For `.mat` files, Matlab® v4 is assumed.

<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.

Example:
```
//Load an IQW file
INP:SEL:FIQ
INP:FILE:PATH 'C:\R_S\Instr\user\data.iqw'
//Define the sample rate
TRAC:IQ:SRAT 10MHz
//Define the measurement time
SENSe:SWEep:TIME  0.001001
//Start the measurement
INIT:IMM
```

Manual operation: See "Select I/Q data file" on page 67

INPut:FILTer:HPASs[:STATe] <State>

Activates an additional internal high-pass filter for RF input signals from 1 GHz to 3 GHz. This filter is used to remove the harmonics of the FSW to measure the harmonics for a DUT, for example.

Requires an additional high-pass filter hardware option.

(Note: for RF input signals outside the specified range, the high-pass filter has no effect. For signals with a frequency of approximately 4 GHz upwards, the harmonics are suppressed sufficiently by the YIG-preselector, if available.)

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

*RST: 0

Example: `INP:FILT:HPAS ON`
Turns on the filter.

Manual operation: See "High Pass Filter 1 to 3 GHz" on page 63

INPut:FILTer:YIG[:STATe] <State>

Enables or disables the YIG filter.

Parameters:

<State> ON | OFF | 0 | 1

Example: `INP:FILT:YIG OFF`
Deactivates the YIG-preselector.

Manual operation: See "YIG-Preselector" on page 63

INPut:IMPedance <Impedance>

Selects the nominal input impedance of the RF input. In some applications, only 50 Ω are supported.

Parameters:

<Impedance> 50 | 75
*RST: 50 Ω
Default unit: OHM

Example: `INP:IMP 75`

Manual operation: See "Impedance" on page 62

INPut:IQ:BALanced[:STATe] <State>

Defines whether the input is provided as a differential signal via all 4 Analog Baseband connectors or as a plain I/Q signal via 2 single-ended lines.

Parameters:

<State> ON | OFF | 1 | 0

ON | 1
Differential

OFF | 0
Single ended

*RST: 1

Example: `INP:IQ:BAL OFF`

Manual operation: See "Input Configuration" on page 65

INPut:SELect <Source>

Selects the signal source for measurements, i.e. it defines which connector is used to input data to the FSW.

If no additional input options are installed, only RF input or file input is supported.

Only selects the RF input or file input. To select the analog baseband input, you have to use INPut<ip>:SELect:BBANalog[:STATe].

For FSW85 models with two RF input connectors, you must select the input connector to configure first using INPut:TYPE.

Parameters:

<Source>

RF
Radio Frequency ("RF INPUT" connector)

FIQ
I/Q data file
(selected by INPut:FILE:PATH on page 197)
Not available for Input2.

AIQ
Analog Baseband signal (only available with optional "Analog Baseband" interface)
Not available for Input2.

RFAiq
RF and analog baseband input active at the same time (for envelope tracking only).

*RST: RF

Example:

`INP:TYPE INP1`
For FSW85 models with two RF input connectors: selects the 1.00 mm RF input connector for configuration.
`INP:SEL RF`

Manual operation: See "I/Q Input File State" on page 66

INPut<ip>:SELect:BBANalog[:STATe] <State>

This command turns simultaneous use of RF input and analog baseband input on and off.

Suffix:

<ip> 1..n

Parameters:

<State> ON | OFF | 1 | 0

Example:

`INP:SEL:BBAN ON`
Turns on the analog baseband input.

Manual operation: See "Enable Parallel BB Capture" on page 65

INPut:TYPE <Input>

The command selects the input path.

Parameters:

<Input>

INPUT1
Selects RF input 1.
1 mm [RF Input] connector

INPUT2
Selects RF input 2.
For FSW85 models with two RF input connectors:
1.85 mm [RF2 Input] connector
For all other models: not available

*RST: INPUT1

Example:

//Select input path
`INP:TYPE INPUT1`

MMEMory:LOAD:IQ:STReam <Channel>

Only available for files that contain more than one data stream from multiple channels: selects the data stream to be used as input for the currently selected channel.

Automatic mode (`MMEMory:LOAD:IQ:STReam:AUTO`) is set to OFF.

Parameters:

<Channel> String containing the channel name.

Example:

```
MMEM:LOAD:IQ:STR?
//Result: 'Channel1','Channel2'
MMEM:LOAD:IQ:STR 'Channel2'
```

Manual operation: See "Selected Channel" on page 67

MMEMory:LOAD:IQ:STReam:AUTO <State>

Only available for files that contain more than one data stream from multiple channels: automatically defines which data stream in the file is used as input for the channel.

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
The data stream specified by `MMEMory:LOAD:IQ:STReam` is used as input for the channel.

ON | 1
The first data stream in the file is used as input for the channel. Applications that support multiple data streams use the first data stream in the file for the first input stream, the second for the second stream etc.

*RST: 1

Manual operation: See "Selected Channel" on page 67

MMEMory:LOAD:IQ:STReam:LIST?

Returns the available channels in the currently loaded input file.

Example:
```
MMEM:LOAD:IQ:STR?
//Result: 'Channel1','Channel2'
```

Usage: Query only

Manual operation: See "Selected Channel" on page 67

TRACe:IQ:FILE:REPetition:COUNt <RepetitionCount>

Determines how often the data stream is repeatedly copied in the I/Q data memory. If the available memory is not sufficient for the specified number of repetitions, the largest possible number of complete data streams is used.

Parameters:
<RepetitionCount> integer

Example: `TRAC:IQ:FILE:REP:COUN 3`

Manual operation: See "File Repetitions" on page 67

5.6.2.2 Remote commands for external frontend control

The following commands are available and required only if the external frontend control option (R&S FSW-K553) is installed.

Further commands for external frontend control described elsewhere:

- `INPut:SELect RF`; see `INPut:SELect` on page 199
- `[SENSe:]FREQuency:CENTer` on page 236
- `DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:RLEVel` on page 238
- `INPut:ATTenuation:AUTO` on page 239
- `INPut:ATTenuation` on page 239
- `INPut:SANalyzer:ATTenuation:AUTO` on page 240

Commands for initial configuration

The following commands are required when you initially set up an external frontend.

[SENSe:]EFRontend:CONNection[:STATe] <State>

Queries the external frontend connection state in the firmware.

Note: to query the physical connection state of the external frontend, use `[SENSe:]EFRontend:CONNection:CSTate?` on page 204.

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
The connection to the frontend is deactivated temporarily. The frontend is thus available for use elsewhere, for example by a signal generator. The measurement settings on the FSW remain untouched.

ON | 1
Frontend connection enabled.
The frontend is reserved for exclusive use by the FSW.

*RST: 0

Example:

```
//Global activation of external frontend
EFR ON
//Configure frontend
EFR:CONN:CONF "FE44S","123.456.789"
//Activate exclusive use of frontend by FSW.
EFR:CONN ON
```

[SENSe:]EFRontend:CONNection:CONFig <Type>, <IPAddress>[, <DeviceID>, <SymbolicName>]

Configures the connection to the external frontend.

Parameters:

<Type> | "FE44S" | "FE50DTR" | "FE170SR" | "FE110SR"

String in double quotes containing the type of frontend to be connected.

<IPAddress> string in double quotes

The IP address or computer name of the frontend connected to the FSW via LAN. The IP address and computer name are indicated on the electronic ink display on the side panel of the frontend.

<DeviceID> string in double quotes

Unique device ID consisting of <type>-<serialnumber>
Not required or relevant for the FSW.

<SymbolicName> string in double quotes

Symbolic name of the external frontend.
Not required or relevant for the FSW.

Example:

```
//Global activation of external frontend
EFR ON
//Configure frontend
EFR:CONN:CONF "FE44S","123.456.789"
//Activate exclusive use of frontend by FSW.
EFR:CONN ON
```

[SENSe:]EFRontend:CONNection:CSTate?

Queries the status of the physical connection to the external frontend.

Return values:

<State> ON | OFF | 0 | 1

OFF | 0
Frontend not connected; connection error

ON | 1
Frontend connected

Usage: Query only

[SENSe:]EFRontend:FREQuency:BAND:COUNt?

Queries the number of frequency bands provided by the selected frontend.

Return values:

<NoBands> integer

Number of frequency bands

Example:

```
//Query number of frequency bands
EFR:FREQ:BAND:COUN?
//Result: 2
```

Usage: Query only

[SENSe:]EFRontend:FREQuency:BAND<b>:LOWer?

Queries the start of the frequency range supported by the selected frontend frequency band.

Suffix:

<b> 1..n
Band for multi-band frontends
Use `[SENSe:]EFRontend:FREQuency:BAND:COUNt?` on page 204 to determine the number of available bands.

Return values:

<StartFreq> Start frequency of the specified band

Example:

```
//Query start frequency of second band
EFR:FREQ:BAND2:LOW?
//Result: 24000000000
```

Usage: Query only

[SENSe:]EFRontend:FREQuency:BAND<b>:UPPer?

Queries the end of the frequency range supported by the selected frontend frequency band.

Suffix:

<b> 1..n
Band for multi-band frontends
Use `[SENSe:]EFRontend:FREQuency:BAND:COUNt?` on page 204 to determine the number of available bands.

Return values:

<StopFreq> End frequency of the specified band

Example:

```
//Query end frequency of second band
EFR:FREQ:BAND2:UPP?
//Result: 44000000000
```

Usage: Query only

[SENSe:]EFRontend:FREQuency:BCONfig:AUTO <State>

Determines whether the frequency band of the external frontend is configured automatically or manually.

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
Uses the frequency band configured by [SENSe:]EFRontend:FREQuency:BCONfig:SELect on page 206.

ON | 1
Configures the frequency band automatically
Currently, auto mode always applies the "IF Low" range.

*RST: 1

Example: //Configures the use of the IF high band manually.

```
EFR:FREQ:BCON:AUTO 0
EFR:FREQ:BCON:SEL "IF HIGH"
```

[SENSe:]EFRontend:FREQuency:BCONfig:LIST?

Returns the intermediate frequency (output) range of the external frontend.

Return values:

<BandConfigs> string

"IF LOW"
(Not for R&S FE170SR/R&S FE110SR frontends.)
A higher intermediate frequency is used on the external frontend, resulting in a higher input frequency at the FSW.

"IF HIGH"
(Not for R&S FE170SR/R&S FE110SR frontends.)
A lower intermediate frequency is used on the external frontend, resulting in a lower input frequency at the FSW.

"Spur Optimized"
The selected IF range avoids unwanted spurious effects.

"EVM Optimized"
The selected IF range provides an optimal EVM result.

Example:

```
EFR:FREQ:BCON:LIST?
//Result: "IF HIGH", "IF LOW"
EFR:FREQ:BCON:SEL "IF HIGH"
```

Usage: Query only

[SENSe:]EFRontend:FREQuency:BCONfig:SELect <BandConfig>

Defines the intermediate frequency (output) range of the external frontend.

Parameters:

<BandConfig>

"IF HIGH"
(R&S FE44S/ R&S FE50DTR)
A higher intermediate frequency is used on the external frontend, resulting in a higher input frequency at the FSW.

"IF LOW"
(R&S FE44S/ R&S FE50DTR)
A lower intermediate frequency is used on the external frontend, resulting in a lower input frequency at the FSW.

"Spur Optimized"
(R&S FE170SR/R&S FE110SR only)
The selected IF range avoids unwanted spurious effects.

"EVM Optimized"
(R&S FE170SR/R&S FE110SR only)
The selected IF range provides an optimal EVM result.

"Shared LO"
(R&S FE170SR/R&S FE110SR only)
Ensures that multiple external frontends (R&S FE170SR/R&S FE170ST or R&S FE110SR/R&S FE110ST) use the same LO frequencies for upconversion and downconversion.

Example:

```
EFR:FREQ:BCON:LIST?
//Result: "IF HIGH", "IF LOW"
EFR:FREQ:BCON:SEL "IF HIGH"
```

[SENSe:]EFRontend:FREQuency:IFRequency:MAXimum?

Queries the maximum used intermediate frequency (IF) for frequency conversion for a frequency span.

Return values:

<IFFrequency> numeric

Example: `EFR:FREQ:IFR:MAX?`

Usage: Query only

[SENSe:]EFRontend:FREQuency:IFRequency:MINimum?

Queries the minimum used intermediate frequency (IF) for frequency conversion for a frequency span.

Return values:

<IFFrequency> numeric

Example: `EFR:FREQ:IFR:MIN?`

Usage: Query only

[SENSe:]EFRontend:FREQuency:IFRequency:SIDeband?

Queries the currently used sideband for frequency conversion.

Return values:

<Sideband> "USB" | "LSB"

"USB"
Upper sideband

"LSB"
Lower sideband

Example: `EFR:FREQ:IFR?`
`EFR:FREQ:IFR:SID?`

Usage: Query only

[SENSe:]EFRontend:FREQuency:IFRequency[:VALue]?

Queries the currently used intermediate frequency (IF) for frequency conversion.

To query the maximum and minimum IF for the selected frequency range, use `[SENSe:]EFRontend:FREQuency:IFRequency:MAXimum?` on page 207 and `[SENSe:]EFRontend:FREQuency:IFRequency:MINimum?` on page 207.

Return values:
<IFFrequency> numeric

Example: `EFR:FREQ:IFR?`

Usage: Query only

[SENSe:]EFRontend:FREQuency:LOSCillator:INPut:FREQuency?

Queries the frequency of the LO input for `[SENSe:]EFRontend:FREQuency:LOSCillator:MODE` `EXT`.

Return values:
<LOInFreq> Default unit: Hz

Example: The external frontend uses the external LO provided at the "LO IN" connector.
`EFR:FREQ:LOSC:MODE EXT`
Query the frequency that the external LO must be provided at.
`EFR:FREQ:LOSC:INP:FREQ?`
`//Result: 10615000000`

Usage: Query only

[SENSe:]EFRontend:FREQuency:LOSCillator:MODE <Type>

Determines whether the external frontend uses its internal LO or an external LO.

Parameters:
<Type> EXTernal | INTernal

EXTernal
Uses the external LO provided at the LO input connector of the external frontend. Query the frequency at which the LO must be input to the external frontend using `[SENSe:]EFRontend:FREQuency:LOSCillator:INPut:FREQuency?` on page 208.

INTernal
Uses the internal LO.

*RST: EXTernal

Example:
```
EFR:FREQ:LOSC:MODE EXT
EFR:FREQ:LOSC:INP:FREQ?
//Result: 10615000000
```

[SENSe:]EFRontend:FREQuency:LOSCillator:OUTPut:FREQuency?

Queries the frequency of the LO output for `[SENSe:]EFRontend:FREQuency:LOSCillator:OUTPut:STATe` ON.

Return values:

<LOOutFreq> Default unit: Hz

Example: The external frontend uses the internal LO and provides it as output to the "LO OUT" connector.
```
EFR:FREQ:LOSC:MODE INT
EFR:FREQ:LOSC:OUTP:STAT ON
```
Query the frequency of the LO output.
```
EFR:FREQ:LOSC:OUTP:FREQ?
//Result: 10615000000
```

Usage: Query only

[SENSe:]EFRontend:FREQuency:LOSCillator:OUTPut:STATe <State>

Enables or disables output of the LO by the external frontend. The output frequency is returned by `[SENSe:]EFRontend:FREQuency:LOSCillator:OUTPut:FREQuency?` on page 209.

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

*RST: 0

Example: The external frontend provides the LO as output at the "LO OUT" connector.
```
EFR:FREQ:LOSC:OUTP:STAT ON
```
Query the frequency of the LO output.
```
EFR:FREQ:LOSC:OUTP:FREQ?
//Result: 10615000000
```

[SENSe:]EFRontend:FREQuency:REFerence <Frequency>

Sets the reference frequency that is used for frequency conversion on the frontend. Depending on the connected type of frontend, different values are available. To determine which reference levels are available, use `[SENSe:]EFRontend:FREQuency:REFerence:LIST?` on page 210.

Parameters:

<Frequency> Default unit: HZ

Example:

```
//Query the available reference levels
EFR:FREQ:REF:LIST?
//Result: 10000000,640000000,1000000000
//Use 640 MHz reference
EFR:FREQ:REF 640000000
```

[SENSe:]EFRontend:FREQuency:REFerence:LIST?

Queries the available reference signals for the connected frontend type.

Return values:

<References> 10000000 | 640000000 | 1000000000

Example:

```
//Query the available reference levels
EFR:FREQ:REF:LIST?
//Result: 10000000,640000000,1000000000
//Use 640 MHz reference
EFR:FREQ:REF 640000000
```

Usage: Query only

[SENSe:]EFRontend:IDN?

Queries the device identifcation information (`*IDN?`) of the frontend.

Return values:

<DevInfo> string without quotes

Rohde&Schwarz,<device type>,<part number>/<serial number>,<firmware version>

Example:

```
EFR:IDN?
//Result: Rohde&Schwarz,FE44S,
1234.5678K00/123456,0.8.0
```

Usage: Query only

[SENSe:]EFRontend:NETWork <IPAddress>, <Subnet>, <DHCP State>

Sets or queries the network information for the frontend.

This information is also indicated on the electronic paper display on the side panel of the device.

Beware that if you change the network setting to DHCP = ON, the connection is aborted and you must re-establish a connection to the frontend (see `[SENSe:]EFRontend:CONNection[:STATe]` on page 203).

Parameters:

<IPAddress> string in double quotes

IP address of the frontend

<Subnet>	string in double quotes Subnet mask of the frontend
<DHCP State>	ON | OFF | 0 | 1 Indicates whether a DHCP server is used. **OFF | 0** DHCP off **ON | 1** DHCP on *RST: 0
Example:	`EFR:NETW?` `//Result: "123.456.78.90","255.255.255.0",ON`

[SENSe:]EFRontend[:STATe] <State>

Enables or disables the general use of an external frontend for the application.

Parameters:

<State>	ON | OFF | 0 | 1 **OFF | 0** The frontend is disconnected. The application adapts the measurement settings to the common settings supported by the FSW. **ON | 1** The FSW allows you to configure and connect an external frontend for the application. The application adapts the available measurement settings to the connected frontend. The channel bar indicates "Inp: ExtFe". *RST: 0
Example:	`EFR ON`

Commands for test, alignment, and diagnosis

The following commands are required to test and optimize the connection after it has initially been set up.

[SENSe:]EFRontend:ALIGnment<ch>:FILE <File>

Selects or queries the touchstone file that contains correction data to compensate for signal losses in the cable occurring at different IF signal frequencies.

Suffix:

<ch> 1..n
Currently irrelevant

Parameters:

<File> string in double quotes

Path and file name of the correction data file. The file must be in `s2p` format.
If the specified file is not found or does not have the correct format, an error message is returned (`-256, "File name not found"`, `-150, "String data error"`).

Example: `EFR:ALIG:FILE "FE44S.s2p"`

[SENSe:]EFRontend:ALIGnment<ch>:STATe <State>

Activates correction of the IF signal due to cable loss from the frontend to the analyzer. Specify the file with correction data using `[SENSe:]EFRontend:ALIGnment<ch>:FILE` on page 211.

Suffix:

<ch> 1..n
Currently irrelevant

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

*RST: 0

[SENSe:]EFRontend:FWUPdate

Updates the firmware on the external frontend. Note that this process can take some time.

Usage: Event

[SENSe:]EFRontend<fe>:SELFtest?

Performs a selftest on the frontend to compare the current performance and characteristic values with the specified values for the frontend.

As a result, the success is returned.

Suffix:
<fe> 1
Connected frontend

Return values:
<Result> **0**
No error

>0
Error

*RST: 0

Example: `EFR:SELF?`
`//Result: 0`

Usage: Query only

[SENSe:]EFRontend<fe>:SELFtest:RESult?

Queries the results of the selftest on the frontend.

Suffix:
<fe> 1
Connected frontend

Return values:
<Result> string containing xml data in double quotes

Example: `EFR:SELF:RES?`

Usage: Query only

Commands for external devices

You can insert additional external devices in the signal path between the DUT and the external frontend, such as preamplifiers or filters. The external frontend must then consider the additional gain or correction values.

[SENSe:]EFRontend[:CHANnel<ch>]:EXTDevice<di>:CORRection:FILE <File>

Defines the correction file to compensate for signal losses by the external devices occurring at different IF signal frequencies.

To query whether the loaded file is valid or not, use [SENSe:]EFRontend[:CHANnel<ch>]:EXTDevice<di>:CORRection:VALid? on page 214.

Suffix:

<ch> 1..n
For future use.

<di> 1..n
irrelevant

Parameters:

<File>

Example:

```
SENS:EFR:CHAN:EXTD:CORR:FILE 'ExtDev_Corr.s2p'
SENS:EFR:CHAN:EXTD:CORR:STAT ON
SENS:EFR:CHAN:EXTD:CORR:FILE:VAL?
```

[SENSe:]EFRontend[:CHANnel<ch>]:EXTDevice<di>:CORRection:STATe <State>

Enables or disables the use of a correction file for the connected external devices. The file is defined by [SENSe:]EFRontend[:CHANnel<ch>]:EXTDevice<di>:CORRection:FILE on page 213.

Suffix:

<ch> 1..n
For future use.

<di> 1..n
irrelevant

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

*RST: 0

Example:

```
SENS:EFR:CHAN:EXTD:CORR:FILE 'ExtDev_Corr.s2p'
SENS:EFR:CHAN:EXTD:CORR:STAT ON
```

[SENSe:]EFRontend[:CHANnel<ch>]:EXTDevice<di>:CORRection:VALid?

Queries whether the loaded correction file for an external device is valid or not.

Suffix:

<ch> 1..n
For future use.

<di> 1..n
irrelevant

Return values:
<Result> ON | OFF | 0 | 1

OFF | 0
File is valid.

ON | 1
File is not valid.

Example:
```
SENS:EFR:CHAN:EXTD:CORR:STAT ON
SENS:EFR:CHAN:EXTD:CORR:FILE 'ExtDev_Corr.s2p'
SENS:EFR:CHAN:EXTD:CORR:VAL?
```

Usage: Query only

[SENSe:]EFRontend[:CHANnel<ch>]:EXTDevice<di>:LIST?

Queries the names of all detected connected devices at the external frontend.

Suffix:
<ch> 1..n
For future use.

<di> 1..n
irrelevant

Return values:
<Devices> Comma-separated list of the names of the devices.

Example:
```
:SENSe1:EFRontend1:CHANnel1:EXTDevice1:LIST?
//Result:
//"fe170_z50","fe170_z01"
```

Usage: Query only

[SENSe:]EFRontend[:CHANnel<ch>]:EXTDevice<di>:REFResh

Scans the ports of the external frontend and refreshes the display with detected information on connected devices. This function is useful after connecting a new device to the frontend.

Suffix:
<ch> 1..n
For future use.

<di> 1..n
irrelevant

Example:
```
SENSe1:EFRontend1:CHANnel:EXTDevice1:REFResh
```

Usage: Event

[SENSe:]EFRontend[:CHANNel<ch>]:EXTDevice<di>:FREQuency:MAXimum?

Queries the upper limit of the supported frequency range of the connected device.

Suffix:

<ch> 1..n
For future use.

<di> 1..n
Connected external device

Return values:

<RFFrequency> numeric value

Maximum frequency

Default unit: Hz

Example:
```
SENS:EFR:EXTD1:FREQ:MAX?
//Result:
170000000000
```

Usage: Query only

[SENSe:]EFRontend[:CHANNel<ch>]:EXTDevice<di>:FREQuency:MINimum?

Queries the lower limit of the supported frequency range of the connected device.

Suffix:

<ch> 1..n
For future use.

<di> 1..n
Connected external device

Return values:

<RFFrequency> numeric value

Minimum frequency

Default unit: Hz

Example:
```
SENS:EFR:EXTD1:FREQ:MIN?
//Result:
110000000000
```

Usage: Query only

[SENSe:]EFRontend[:CHANNel<ch>]:EXTDevice<di>:GAIN?

Queries the gain provided by a connected external amplifier.

Suffix:

<ch> 1..n
For future use.

<di> 1..n
Connected external device

Return values:

<Gain> numeric value

Default unit: dB

Example:
```
SENSe1:EFRontend1:CHANnel1:EXTDevice1:GAIN?
//Result:
//10
```

Usage: Query only

[SENSe:]EFRontend[:CHANnel<ch>]:EXTDevice<di>:NAME?

Queries the name of the external device for reference.

Suffix:

<ch> 1..n
For future use.

<di> 1..n
Connected external device

Return values:

<Type> string

Example:
```
SENS:EFR:CHAN:EXTD1:NAME?
//Result:
//'fe170_z50'
```

Usage: Query only

[SENSe:]EFRontend[:CHANnel<ch>]:EXTDevice<di>:TYPE?

Queries the type of the detected device for reference.

Suffix:

<ch> 1..n
For future use.

<di> 1..n
Connected external device

Return values:

<Type> 'Amplifier' | 'Filter'

Example:
```
SENSe1:EFRontend1:CHANnel1:EXTDevice1:TYPE?
//Result:
//'Filter'
```

Usage: Query only

Format of selftest result file

As a result of the selftest, an XML file with test details is created. It contains the following information in the specified order. Mandatory elements and attributes are indicated in bold font.

Element	Attributes	Description
<Sequence>		Main element
	Name	="Selftest"
	Description	Optional description of the test process
	FirmwareVersion	Firmware version of the controlling instrument (FSW)
	FrontendLibrary-Version	Version of the `control.dll` with the format `x.y.z`
	FrontendServerVer-sion	Version of the RRH server with the format `x.y.z` (FE44A only)
	Date	Date the selftest was performed, with the format `dd/mm/yyyy`
	Time	Time the selftest was performed, with the format `hh:mm:ss`
	State	Test result state, combined result of all <SequenceCate-gory>s: "`PASSED`"/ "`FAILED`"
	Version	For internal use.
	Comment	Optional comment on the test process
<SequenceCate-gory>		Set of test steps
	Name	Name of the test sequence, e.g. "Frontend voltages"
	Description	Optional description of the test sequence
	State	Test sequence result state, combined result of all <Sequence-Step>s: "`PASSED`"/ "`FAILED`"
	Version	For internal use.
	Comment	Optional comment on the test sequence
<SequenceStep>		Subelement of **<SequenceCategory>** for an individual test step
	Name	Name of the individual test step, e.g. FE1_3V3
	Description	Optional description of the test step
	LimitLow	Optional: lower limit to be checked
	LimitHigh	Optional: upper limit to be checked
	MeasValue	Optional: measured value
	Unit	Optional: unit of the measured value
	State	Test step result state:"`PASSED`"/ "`FAILED`"
	Version	For internal use.
	Comment	Optional comment on the test step

Example for selftest result xml file

```
<?xml version="1.0" encoding="UTF-8"?>
<Sequence xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
        xsi:noNamespaceSchemaLocation="SelfTest_Schema.xsd">
<Name>DeviceCheck</Name>
<FirmwareVersion>1.50-21.1.1.0 Beta</FirmwareVersion>
<FrontendLibraryVersion>0.8.0</FrontendLibraryVersion>
<Date>15/01/2021</Date>
<Time>08:51:35</Time>
<State>FAILED</State>
<Version>1.0.0</Version>
<SequenceCategory>
 <Name>Frontend Voltages</Name>
 <Description>test description</Description>
 <State>FAILED</State>
 <Version>1.0.0</Version>
 <Type>Diagnose</Type>
 <SequenceStep>
  <Name>FE1_3V3</Name>
  <LimitLow>3.000</LimitLow>
  <LimitHigh>3.600</LimitHigh>
  <MeasValue>3.311</MeasValue>
  <State>PASSED</State>
  <Version>1.0.0</Version>
 </SequenceStep>
 <SequenceStep>
  <Name>FE1_5V</Name>
  <LimitLow>4.250</LimitLow>
  <LimitHigh>5.750</LimitHigh>
  <MeasValue>5.027</MeasValue>
  <State>PASSED</State>
  <Version>1.0.0</Version>
 </SequenceStep>
 <SequenceStep>
  <Name>FE1_6V5</Name>
  <LimitLow>6.000</LimitLow>
  <LimitHigh>7.000</LimitHigh>
  <MeasValue>5.893</MeasValue>
  <State>FAILED</State>
  <Version>1.0.0</Version>
 </SequenceStep>
 <SequenceStep>
  <Name>FE1_M3V3</Name>
  <LimitLow>-3.600</LimitLow>
  <LimitHigh>-3.000</LimitHigh>
  <MeasValue>3.347</MeasValue>
  <State>FAILED</State>
  <Version>1.0.0</Version>
 </SequenceStep>
</SequenceCategory>
<SequenceCategory>
 <Name>Frontend Temperature</Name>
 <Description>test description</Description>
```

```
  <State>PASSED</State>
  <Version>1.0.0</Version>
  <Type>Diagnose</Type>
  <SequenceStep>
   <Name>TEMP_FE1</Name>
   <LimitLow>0.000</LimitLow>
   <LimitHigh>60.000</LimitHigh>
   <MeasValue>39.300</MeasValue>
   <State>PASSED</State>
   <Version>1.0.0</Version>
  </SequenceStep>
 </SequenceCategory>
 <SequenceCategory>
  <Name>Synthesizer Voltage</Name>
  <Description>test description</Description>
  <State>PASSED</State>
  <Version>1.0.0</Version>
  <Type>Diagnose</Type>
  <SequenceStep>
   <Name>SYNTH_3V4</Name>
   <LimitLow>3.200</LimitLow>
   <LimitHigh>3.910</LimitHigh>
   <MeasValue>3.576</MeasValue>
   <State>PASSED</State>
   <Version>1.0.0</Version>
  </SequenceStep>
  <SequenceStep>
   <Name>SYNTH_5V4_SYN</Name>
   <LimitLow>4.860</LimitLow>
   <LimitHigh>5.940</LimitHigh>
   <MeasValue>5.405</MeasValue>
   <State>PASSED</State>
   <Version>1.0.0</Version>
  </SequenceStep>
  <SequenceStep>
   <Name>SYNTH_5V4_YIG</Name>
   <LimitLow>4.860</LimitLow>
   <LimitHigh>5.940</LimitHigh>
   <MeasValue>5.438</MeasValue>
   <State>PASSED</State>
   <Version>1.0.0</Version>
  </SequenceStep>
  <SequenceStep>
   <Name>SYNTH_M5V</Name>
   <LimitLow>-5.500</LimitLow>
   <LimitHigh>-4.500</LimitHigh>
   <MeasValue>-4.948</MeasValue>
   <State>PASSED</State>
   <Version>1.0.0</Version>
  </SequenceStep>
```

```
  <SequenceStep>
   <Name>SYNTH_REF5V</Name>
   <LimitLow>4.500</LimitLow>
   <LimitHigh>5.500</LimitHigh>
   <MeasValue>5.031</MeasValue>
   <State>PASSED</State>
   <Version>1.0.0</Version>
  </SequenceStep>
 </SequenceCategory>
 <SequenceCategory>
  <Name>Supply Voltage</Name>
  <Description>test description</Description>
  <State>PASSED</State>
  <Version>1.0.0</Version>
  <Type>Diagnose</Type>
  <SequenceStep>
   <Name>SUPPLY_12V</Name>
   <LimitLow>10.800</LimitLow>
   <LimitHigh>13.200</LimitHigh>
   <MeasValue>11.909</MeasValue>
   <State>PASSED</State>
   <Version>1.0.0</Version>
  </SequenceStep>
  <SequenceStep>
   <Name>SUPPLY_3V3D</Name>
   <LimitLow>2.970</LimitLow>
   <LimitHigh>3.630</LimitHigh>
   <MeasValue>3.318</MeasValue>
   <State>PASSED</State>
   <Version>1.0.0</Version>
  </SequenceStep>
  <SequenceStep>
   <Name>SUPPLY_4V</Name>
   <LimitLow>3.650</LimitLow>
   <LimitHigh>4.460</LimitHigh>
   <MeasValue>4.053</MeasValue>
   <State>PASSED</State>
   <Version>1.0.0</Version>
  </SequenceStep>
  <SequenceStep>
   <Name>SUPPLY_6V</Name>
   <LimitLow>5.400</LimitLow>
   <LimitHigh>6.600</LimitHigh>
   <MeasValue>6.076</MeasValue>
   <State>PASSED</State>
   <Version>1.0.0</Version>
  </SequenceStep>
  <SequenceStep>
   <Name>SUPPLY_M5V</Name>
   <LimitLow>-6.050</LimitLow>
```

```
    <LimitHigh>-4.950</LimitHigh>
    <MeasValue>-5.507</MeasValue>
    <State>PASSED</State>
    <Version>1.0.0</Version>
  </SequenceStep>
 </SequenceCategory>
</Sequence>
```

Programming example: Configuring an external frontend

The following example describes how to configure RF frontend settings in remote operation.

```
// Prepare the instrument
// Preset
*RST
// Create new IQ-Analyzer channel
:INST:SEL IQ
// Enable 640MHz Reference
:ROSC:O640 ON

//Enable general use of external frontend
SENSe:EFRontend:STATe ON
//Configure connection to ext. FE named "FE44S-1000826"
SENSe:EFRontend:CONNection:CONFig "FE44S","FE44S-1000826"
//Activate exclusive use of frontend by FSW.
SENSe:EFRontend:CONNection:STATe ON

//For demonstration purposes only: assign a static IP address
SENSe:EFRontend:NETWork "123.456.7.8", "255.255.255.00", OFF

// Query information about the connected RF frontend.
SENSe:EFRontend:CONNection:CSTATe?
// Response: 1 (connected)
SENSe:EFRontend:IDN?
////Result: Rohde&Schwarz,FE44S,1234.5678K00/123456,0.8.0

// Specify frontend settings
//Query available intermediate frequency bands
SENSe:EFRontend:FREQuency:BCONfig:LIST?
//Result: "IF HIGH", "IF LOW"
//Use high IF
SENSe:EFRontend:FREQuency:BCONfig:SELect "IF HIGH"
//Query used intermediate frequency
SENSe:EFRontend:FREQuency:IFRequency?
//Result: 8.595000000

//Query available reference frequencies
SENSe:EFRontend:FREQuency:REFerence:LIST?
//Result: 10000000,640000000,1000000000
```

```
//Use 640 MHz reference
SENSe:EFRontend:FREQuency:REFerence 640000000

// Query ranges of the operating frequency band.
SENSe:EFRontend:FREQuency:BAND1:LOWer?
// Response in Hz: "24000000000" (= 24 GHz)
SENSe:EFRontend:FREQuency:BAND1:UPPer?
// Response in Hz: "44000000000" (= 44 GHz)

// Add cable correction data by loading an *.s2p file.
SENSe:EFRontend:ALIGnment:FILE "C:\R_S\Instr\user\external_frontends\FE44S\
touchstonefiles\if_default_cable_1347_7552_00.s2p"
SENSe:EFRontend:ALIGnment:STATe ON

//Update FW version on frontend (only available if external frontend firmware
//is incompatible to FSW firmware)
SENSe:EFRontend:FWUPdate
//Perform a selftest on the frontend and query results
SENSe:EFRontend:SELFtest?
//Result: 0 (no errors)
SENSe:EFRontend:SELFtest:RESult?
//Result: "<?xml version="1.0" encoding="UTF-8"?>
//<Sequence xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
//xsi:noNamespaceSchemaLocation="
//<Name>DeviceCheck</Name>
//<FirmwareVersion>1.50-21.1.1.0 Beta</FirmwareVersion>
//..."
```

Programming example 2: Configuring an external frontend with a connected amplifier

The following example demonstrates how to configure an R&S FE170SR external frontend with a connected amplifier.

Some commands may not be necessary as they reflect the default settings; however, they are included to demonstrate the command usage.

```
// Prepare the instrument
// Preset
*RST
// Create new IQ-Analyzer channel
:INST:SEL IQ
// Enable 640MHz Reference
:ROSC:O640 ON

//Enable general use of external frontend
SENSe:EFRontend:STATe ON
//Configure connection to ext. FE named "FE170SR-123456"
SENSe:EFRontend:CONNection:CONFig "FE170SR","FE170SR-123456"
//Activate exclusive use of frontend by FSW.
SENSe:EFRontend:CONNection:STATe ON
```

```
// Query information about the connected RF frontend.
SENSe:EFRontend:CONNection:CSTAte?
// Response: 1 (connected)
SENSe:EFRontend:IDN?
////Result: Rohde&Schwarz,FE170SR,1234.5678K00/123456,0.8.0

// Specify frontend settings
//Query available intermediate frequency bands
SENSe:EFRontend:FREQuency:BCONfig:LIST?
//Result: "Spur Optimized","EVM Optimized"
//Use spur optimized
SENSe:EFRontend:FREQuency:BCONfig:SELect "Spur Optimized"
//Query used intermediate frequency
SENSe:EFRontend:FREQuency:IFRequency?
//Result: 9940000000


//Query available reference frequencies
SENSe:EFRontend:FREQuency:REFerence:LIST?
//Result: 10000000,640000000,1000000000
//Use 1 GHz reference
SENSe:EFRontend:FREQuency:REFerence 1000000000

// Query ranges of the operating frequency band.
SENSe:EFRontend:FREQuency:BAND1:LOWer?
// Response in Hz: "110000000000" (= 110 GHz)
SENSe:EFRontend:FREQuency:BAND1:UPPer?
// Response in Hz: "170000000000" (= 170 GHz)

// Add cable correction data by loading an *.s2p file.
SENSe:EFRontend:ALIGnment:FILE "C:\R_S\Instr\user\external_frontends\FE170SR\
touchstonefiles\if_default_cable_1348_3850_00.s2p"
SENSe:EFRontend:ALIGnment:STATe ON

//Configure external amplifier named "fe170_z50" connected to frontend
//Refresh and query the list of detected external devices
SENSe:EFRontend:EXTDevice:REFResh
SENSe:EFRontend:EXTDevice:LIST?
//Result:
//"fe170_z50"
//Query the type of the external device
SENSe:EFRontend:EXTDevice:TYPE?
//Result:
//'Amplifier'
//Query the supported frequency range of the amplifier
SENSe:EFRontend:EXTDevice1:FREQuency:MINimum?
//Result:
//110000000000
SENSe:EFRontend:EXTDevice1:FREQuency:MAXimum?
//Result:
//170000000000
//Query the gain of the amplifier
SENSe:EFRontend:EXTDevice1:GAIN?
```

```
//Result:
//10
//Only for demonstration purposes:
//Load and enable a correction file for the amplifier.
//The correction file is automatically considered if the IX cable is used to connect
//the external device to the frontend.
SENSe:EFRontend:EXTDevice1:CORRection:FILE 'C:\temp\ExtDev_Corr.s2p'
SENSe:EFRontend:EXTDevice1:CORRection:STATe ON
//Check if the loaded file is valid
SENSe:EFRontend:EXTDevice1:CORRection:VALid?
```

5.6.3 Power sensor measurements

The following commands describe how to work with power sensors.

These commands require the use of a Rohde & Schwarz power sensor. For a list of supported sensors, see the specifications document.

5.6.3.1 Configuring power sensor measurements

SYSTem:COMMunicate:RDEVice:PMETer<p>:CONFigure:AUTO[:STATe] <State>

Turns automatic assignment of a power sensor to the power sensor index on and off.

Suffix:
<p> Power sensor index

Parameters:
<State> ON | OFF | 0 | 1
*RST: 1

Example: `SYST:COMM:RDEV:PMET:CONF:AUTO OFF`

Manual operation: See "Select" on page 76

SYSTem:COMMunicate:RDEVice:PMETer<p>:COUNt?

Queries the number of power sensors currently connected to the FSW.

Suffix:
<p> Power sensor index

Return values:
<NumberSensors> Number of connected power sensors.

Example: `SYST:COMM:RDEV:PMET:COUN?`

Usage: Query only

Manual operation: See "Select" on page 76

SYSTem:COMMunicate:RDEVice:PMETer<p>:DEFine <Placeholder>, <Type>, <Interface>, <SerialNo>

Assigns the power sensor with the specified serial number to the selected power sensor index (configuration).

The query returns the power sensor type and serial number of the sensor assigned to the specified index.

Suffix:
<p> Power sensor index

Parameters:
<Placeholder> Currently not used

<Type> Detected power sensor type, e.g. "NRP-Z81".

<Interface> Interface the power sensor is connected to; always "USB"

<SerialNo> Serial number of the power sensor assigned to the specified index

Example: `SYST:COMM:RDEV:PMET2:DEF '','NRP-Z81','', '123456'`
Assigns the power sensor with the serial number '123456' to the configuration "Power Sensor 2".
`SYST:COMM:RDEV:PMET2:DEF?`
Queries the sensor assigned to "Power Sensor 2".
Result:
`'','NRP-Z81','USB','123456'`
The NRP-Z81 power sensor with the serial number '123456' is assigned to the "Power Sensor 2".

Manual operation: See "Select" on page 76

CALibration:PMETer<p>:ZERO:AUTO ONCE

Zeroes the power sensor.

Note that you have to disconnect the signals from the power sensor input before you start to zero the power sensor. Otherwise, results are invalid.

Suffix:
<p> Power sensor index

Example: `CAL:PMET2:ZERO:AUTO ONCE;*WAI`
Starts zeroing the power sensor 2 and delays the execution of further commands until zeroing is concluded.

Usage: Event

Manual operation: See "Zeroing Power Sensor" on page 76

CALCulate<n>:PMETer<p>:RELative[:MAGNitude] <RefValue>

Defines the reference value for relative measurements.

Suffix:
<n> Window

<p> Power sensor index

Parameters:
<RefValue> Range: -200 dBm to 200 dBm
*RST: 0
Default unit: DBM

Example: `CALC:PMET2:REL -30`
Sets the reference value for relative measurements to -30 dBm for power sensor 2.

Manual operation: See "Reference Value" on page 77

CALCulate<n>:PMETer<p>:RELative[:MAGNitude]:AUTO ONCE

Sets the current measurement result as the reference level for relative measurements.

Suffix:

<n>	Window
<p>	Power sensor index

Example: `CALC:PMET2:REL:AUTO ONCE`
Takes the current measurement value as reference value for relative measurements for power sensor 2.

Usage: Event

Manual operation: See "Setting the Reference Level from the Measurement Meas -> Ref" on page 77

CALCulate<n>:PMETer<p>:RELative:STATe <State>

Turns relative power sensor measurements on and off.

Suffix:

<n>	Window
<p>	Power sensor index

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

Example: `CALC:PMET2:REL:STAT ON`
Activates the relative display of the measured value for power sensor 2.

FETCh:PMETer<p>?

Queries the results of power sensor measurements.

Suffix:

<p> Power sensor index

Usage: Query only

READ:PMETer<p>?

Initiates a power sensor measurement and queries the results.

Suffix:

<p> Power sensor index

Usage: Query only

[SENSe:]PMETer<p>:DCYCle[:STATe] <State>

Turns the duty cycle correction on and off.

Suffix:
<p> Power sensor index

Parameters:
<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

Example: `PMET2:DCYC:STAT ON`

Manual operation: See "Duty Cycle" on page 78

[SENSe:]PMETer<p>:DCYCle:VALue <Percentage>

Defines the duty cycle for the correction of pulse signals.

The power sensor uses the duty cycle in combination with the mean power to calculate the power of the pulse.

Suffix:
<p> Power sensor

Parameters:
<Percentage> Range: 0.001 to 99.999
*RST: 99.999
Default unit: %

Example: `PMET2:DCYC:STAT ON`
Activates the duty cycle correction.
`PMET2:DCYC:VAL 0.5`
Sets the correction value to 0.5%.

Manual operation: See "Duty Cycle" on page 78

[SENSe:]PMETer<p>:FREQuency <Frequency>

Defines the frequency of the power sensor.

Suffix:
<p> Power sensor index

Parameters:
<Frequency> The available value range is specified in the specifications document of the power sensor in use.
*RST: 50 MHz
Default unit: HZ

Example: `PMET2:FREQ 1GHZ`
Sets the frequency of the power sensor to 1 GHz.

Manual operation: See "Frequency Manual" on page 76

[SENSe:]PMETer<p>:FREQuency:LINK <Coupling>

Selects the frequency coupling for power sensor measurements.

Suffix:
<p> Power sensor index

Parameters:
<Coupling>

CENTer
Couples the frequency to the center frequency of the analyzer

MARKer1
Couples the frequency to the position of marker 1

OFF
Switches the frequency coupling off

*RST: CENTer

Example: `PMET2:FREQ:LINK CENT`
Couples the frequency to the center frequency of the analyzer

Manual operation: See "Frequency Coupling" on page 76

[SENSe:]PMETer<p>:MTIMe <Duration>

Selects the duration of power sensor measurements.

Suffix:
<p> Power sensor index

Parameters:
<Duration> SHORt | NORMal | LONG

*RST: NORMal

Example: `PMET2:MTIM SHOR`
Sets a short measurement duration for measurements of stationary high power signals for the selected power sensor.

Manual operation: See "Meas Time/Average" on page 77

[SENSe:]PMETer<p>:MTIMe:AVERage:COUNt <NumberReadings>

Sets the number of power readings included in the averaging process of power sensor measurements.

Extended averaging yields more stable results for power sensor measurements, especially for measurements on signals with a low power, because it minimizes the effects of noise.

Suffix:
<p> Power sensor index

Parameters:
<NumberReadings> An average count of 0 or 1 performs one power reading.
Range: 0 to 256
Increment: binary steps (1, 2, 4, 8, ...)

Example:
`PMET2:MTIM:AVER ON`
Activates manual averaging.
`PMET2:MTIM:AVER:COUN 8`
Sets the number of readings to 8.

Manual operation: See "Average Count (Number of Readings)" on page 78

[SENSe:]PMETer<p>:MTIMe:AVERage[:STATe] <State>

Turns averaging for power sensor measurements on and off.

Suffix:
<p> Power sensor index

Parameters:
<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

Example:
`PMET2:MTIM:AVER ON`
Activates manual averaging.

Manual operation: See "Meas Time/Average" on page 77

[SENSe:]PMETer<p>:ROFFset[:STATe] <State>

Includes or excludes the reference level offset of the analyzer for power sensor measurements.

Suffix:
<p> Power sensor index

Parameters:
<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

Example:
`PMET2:ROFF OFF`
Takes no offset into account for the measured power.

Manual operation: See "Use Ref Level Offset" on page 77

[SENSe:]PMETer<p>:SOFFset <SensorOffset>

Takes the specified offset into account for the measured power. Only available if `[SENSe:]PMETer<p>:ROFFset[:STATe]` is disabled.

Suffix:
<p> Power sensor index

Parameters:
<SensorOffset> Default unit: DB

Example: `PMET2:SOFF 0.001`

Manual operation: See "Sensor Level Offset" on page 78

[SENSe:]PMETer<p>[:STATe] <State>

Turns a power sensor on and off.

Suffix:
<p> Power sensor index

Parameters:
<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

Example: `PMET1 ON`
Switches the power sensor measurements on.

Manual operation: See "Select" on page 76

[SENSe:]PMETer:LEVel:CORRection:APPLy <State>

Sets and queries the toggle to apply the auto level correction.

Parameters:
<State> ON | OFF | 0 | 1

OFF | 0

ON | 1

Manual operation: See "Apply Auto Level Correction" on page 75

CALCulate:PMETer:LEVel:CORRection

Calculates the level correction for power sensors.

Example: `CALCulate:PMETer:LEVel:CORRection`

Usage: Event

Manual operation: See "Apply Auto Level Correction" on page 75

TRIGger[:SEQuence]:SOURce <Source>

Selects the trigger source.

Note on external triggers:

If a measurement is configured to wait for an external trigger signal in a remote control program, remote control is blocked until the trigger is received and the program can continue. Make sure that this situation is avoided in your remote control programs.

Parameters:

<Source>

IMMediate
Free Run

EXTernal
Trigger signal from the "Trigger Input" connector.

EXT2
Trigger signal from the "Trigger Input/Output" connector.
For FSW85 models, Trigger 2 is not available due to the second RF input connector on the front panel. The trigger signal is taken from the "Trigger Input/Output" connector on the rear panel.
Note: Connector must be configured for "Input".

EXT3
Trigger signal from the "TRIGGER 3 INPUT/ OUTPUT" connector.
Note: Connector must be configured for "Input".

BBPower
Baseband power
For input from the optional "Analog Baseband" interface.
For input from the optional "Digital Baseband" interface.

PSEN
External power sensor

*RST: IMMediate

Example:

`TRIG:SOUR EXT`
Selects the external trigger input as source of the trigger signal

Manual operation: See "Using the power sensor as an external trigger" on page 78

UNIT<n>:PMETer<p>:POWer <Unit>

Selects the unit for absolute power sensor measurements.

Suffix:

<n> irrelevant

<p> Power sensor index

Parameters:

<Unit> DBM | WATT | W | DB | PCT

*RST: DBM

Example:

`UNIT:PMET:POW DBM`

Manual operation: See "Unit/Scale" on page 77

UNIT<n>:PMETer<p>:POWer:RATio <Unit>

Selects the unit for relative power sensor measurements.

Suffix:

<n>	irrelevant
<p>	Power sensor index

Parameters:

<Unit>	DB | PCT *RST: DB

Example: `UNIT:PMET:POW:RAT DB`

Manual operation: See "Unit/Scale" on page 77

5.6.3.2 Triggering with power sensors

[SENSe:]PMETer<p>:TRIGger:DTIMe <Time>

Defines the time period that the input signal has to stay below the IF power trigger level before the measurement starts.

Suffix:

<p>	Power sensor index

Parameters:

<Time>	Range: 0 s to 1 s Increment: 100 ns *RST: 100 µs Default unit: S

Example: `PMET2:TRIG:DTIMe 0.001`

[SENSe:]PMETer<p>:TRIGger:HOLDoff <Holdoff>

Defines the trigger holdoff for external power triggers.

Suffix:

<p>	Power sensor index

Parameters:

<Holdoff> Time period that has to pass between the trigger event and the start of the measurement, in case another trigger event occurs.

Range: 0 s to 1 s
Increment: 100 ns
*RST: 0 s
Default unit: S

Example: `PMET2:TRIG:HOLD 0.1`
Sets the holdoff time of the trigger to 100 ms

Manual operation: See "Trigger Holdoff" on page 79

[SENSe:]PMETer<p>:TRIGger:HYSTeresis <Hysteresis>

Defines the trigger hysteresis for external power triggers.

The hysteresis in dB is the value the input signal must stay below the IF power trigger level to allow a trigger to start the measurement.

Suffix:

<p> Power sensor index

Parameters:

<Hysteresis> Range: 3 dB to 50 dB
Increment: 1 dB
*RST: 0 dB
Default unit: DB

Example: `PMET2:TRIG:HYST 10`
Sets the hysteresis of the trigger to 10 dB.

Manual operation: See "Hysteresis" on page 78

[SENSe:]PMETer<p>:TRIGger:LEVel <Level>

Defines the trigger level for external power triggers.

Suffix:

<p> Power sensor index

Parameters:

<Level> -20 to +20 dBm

Range: -20 dBm to 20 dBm
*RST: -10 dBm
Default unit: DBM

Example: `PMET2:TRIG:LEV -10 dBm`
Sets the level of the trigger

Manual operation: See "External Trigger Level" on page 78

[SENSe:]PMETer<p>:TRIGger:SLOPe <Edge>

Selects the trigger condition for external power triggers.

Suffix:
<p> Power sensor index

Parameters:
<Edge>

POSitive
The measurement starts in case the trigger signal shows a positive edge.

NEGative
The measurement starts in case the trigger signal shows a negative edge.

*RST: POSitive

Example: `PMET2:TRIG:SLOP NEG`

Manual operation: See "Slope" on page 79

[SENSe:]PMETer<p>:TRIGger[:STATe] <State>

Turns the external power trigger on and off.

Suffix:
<p> Power sensor index

Parameters:
<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

Example: `PMET2:TRIG ON`
Switches the external power trigger on

Manual operation: See "Using the power sensor as an external trigger" on page 78

5.6.4 Configuring the frequency

[SENSe:]FREQuency:CENTer <Frequency>

Defines the center frequency.

Parameters:

<Frequency> For the allowed range and f_{max}, refer to the specifications document.

*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 68

[SENSe:]FREQuency:CENTer:STEP <StepSize>

Defines the center frequency step size.

Parameters:

<StepSize> For f_{max}, refer to the specifications document.

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 68

[SENSe:]FREQuency:OFFSet <Offset>

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 68

5.6.5 Defining level characteristics

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:RLEVel <ReferenceLevel>

Defines the reference level (for all traces in all windows).

With a reference level offset ≠ 0, the value range of the reference level is modified by the offset.

Suffix:

<n> irrelevant

<w> subwindow
Not supported by all applications

<t> irrelevant

Parameters:

<ReferenceLevel> The unit is variable.

Range: see specifications document
*RST: 0 dBm
Default unit: DBM

Example: `DISP:TRAC:Y:RLEV -60dBm`

Manual operation: See "Reference Level" on page 69

DISPlay[:WINDow<n>][:SUBWindow<w>]:TRACe<t>:Y[:SCALe]:RLEVel:OFFSet <Offset>

Defines a reference level offset (for all traces in all windows).

Suffix:

<n> irrelevant

<w> subwindow
Not supported by all applications

<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 70

INPut:ATTenuation <Attenuation>

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.

If an external frontend is active (see `[SENSe:]EFRontend[:STATe]` on page 211), you can configure the attenuation of the external frontend and the analyzer separately. See also `INPut:SANalyzer:ATTenuation:AUTO` on page 240 and `INPut:SANalyzer:ATTenuation` on page 240.

Parameters:

<Attenuation> Range: see specifications document
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 72

INPut:ATTenuation:AUTO <State>

Couples or decouples the attenuation to the reference level. Thus, when the reference level is changed, the FSW determines the signal level for optimal internal data processing and sets the required attenuation accordingly.

If an external frontend is active (see `[SENSe:]EFRontend[:STATe]` on page 211), you can configure the attenuation of the external frontend and the analyzer separately. See also `INPut:SANalyzer:ATTenuation:AUTO` on page 240 and `INPut:SANalyzer:ATTenuation` on page 240.

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 72

INPut:SANalyzer:ATTenuation <Attenuation>

Configures attenuation at the analyzer input for an active external frontend manually.

Parameters:

<Attenuation> Range: see specifications document
Increment: 1 dB
Default unit: DB

Manual operation: See "Attenuation Mode / Value" on page 72

INPut:SANalyzer:ATTenuation:AUTO <State>

Enables or disables automatic configuration of attenuation at the analyzer input for an active external frontend.

By default, the attenuation settings are applied at the input of the external frontend.

See `INPut:ATTenuation:AUTO` on page 239 and `INPut:ATTenuation` on page 239.

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
Auto mode for analyzer attenuation is disabled. Allows you to configure attenuation at the analyzer using `INPut:SANalyzer:ATTenuation` on page 240.

ON | 1
Auto mode for analyzer attenuation is enabled. No attenuation is configured at the analyzer.

*RST: 0

Example:

```
//Enable external frontend
EFR ON
//Query the currently configured RF attenuation
INP:ATT?
//Result: 10 dB
//Disable auto mode for analyzer attenuation
INP:SAN:ATT:AUTO OFF
//Configure 10 dB attenuation at the analyzer
INP:SAN:ATT 10
//Query the currently configured RF attenuation at the ext. FE
INP:ATT?
//Result: 0 dB
```

Manual operation: See "Attenuation Mode / Value" on page 72

INPut:ATTenuation:AUTO:MODE <OptMode>

Selects the priority for signal processing *after* the RF attenuation has been applied.

Parameters:

<OptMode> LNOise | LDIStortion

LNOise
Optimized for high sensitivity and low noise levels

LDIStortion
Optimized for low distortion by avoiding intermodulation

*RST: LDIStortion (WLAN application: LNOise)

Example: `INP:ATT:AUTO:MODE LNO`

Manual operation: See "Optimization" on page 72

INPut:EATT <Attenuation>

Defines an electronic attenuation manually. Automatic mode must be switched off (`INP:EATT:AUTO OFF`, see `INPut:EATT:AUTO` on page 241).

If the current reference level is not compatible with an attenuation that has been set manually, the command also adjusts the reference level.

Parameters:

<Attenuation> attenuation in dB

Range: see specifications document
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 73

INPut:EATT:AUTO <State>

Turns automatic selection of the electronic attenuation on and off.

If on, electronic attenuation reduces the mechanical attenuation whenever possible.

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 73

INPut:EATT:STATe <State>

Turns the electronic attenuator on and off.

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 73

INPut:EGAin[:STATe] <State>

Before this command can be used, the external preamplifier must be connected to the FSW. See the preamplifier's documentation for details.

When activated, the FSW automatically compensates the magnitude and phase characteristics of the external preamplifier in the measurement results.

Note that when an optional external preamplifier is activated, the internal preamplifier is automatically disabled, and vice versa.

For FSW85 models with two RF inputs, you must enable correction from the external preamplifier for each input individually. Correction cannot be enabled for both inputs at the same time.

When deactivated, no compensation is performed even if an external preamplifier remains connected.

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
No data correction is performed based on the external preamplifier

ON | 1
Performs data corrections based on the external preamplifier

*RST: 0

Example: `INP:EGA ON`

Manual operation: See "Ext. PA Correction" on page 71

INPut:GAIN:STATe <State>

Turns the internal preamplifier on and off. It requires the optional preamplifier hardware.

Note that if an optional external preamplifier is activated, the internal preamplifier is automatically disabled, and vice versa.

Is not available for input from the optional "Digital Baseband" interface.

For FSW85 models, no preamplifier is available.

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

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

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`
`INP:GAIN:VAL 15`
Switches on 15 dB preamplification.

Manual operation: See "Preamplifier" on page 70

INPut:GAIN[:VALue] <Gain>

Selects the "gain" if the preamplifier is activated (`INP:GAIN:STAT ON`, see `INPut:GAIN:STATe` on page 242).

The command requires the additional preamplifier hardware option.

Parameters:

<Gain> For all FSW models except for FSW85, the following settings are available:
15 dB and 30 dB
All other values are rounded to the nearest of these two.
30 dB
For older FSW43/FSW50/FSW67 models, the input signal is always ampilfied by about 30 dB when the preamplifier is active.
For FSW85 models, no preamplifier is available.

Default unit: DB

Example:
`INP:GAIN:STAT ON`
`INP:GAIN:VAL 30`
Switches on 30 dB preamplification.

Manual operation: See "Preamplifier" on page 70

INPut:IQ:FULLscale:AUTO <State>

Defines whether the full scale level (i.e. the maximum input power on the Baseband Input connector) is defined automatically according to the reference level, or manually.

Parameters:

<State>

ON | 1
Automatic definition

OFF | 0
Manual definition according to `INPut:IQ:FULLscale[:LEVel]` on page 244

*RST: 1

Example: `INP:IQ:FULL:AUTO OFF`

INPut:IQ:FULLscale[:LEVel] <PeakVoltage>

Defines the peak voltage at the Baseband Input connector if the full scale level is set to manual mode (see `INPut:IQ:FULLscale:AUTO` on page 244).

Parameters:

<PeakVoltage> 0.25 V | 0.5 V | 1 V | 2 V

Peak voltage level at the connector.
For probes, the possible full scale values are adapted according to the probe's attenuation and maximum allowed power.

*RST: 1V
Default unit: V

Example: `INP:IQ:FULL 0.5V`

Manual operation: See "Full Scale Level" on page 70

[SENSe:]IQ:WBANd <State>

Selects the signal path for signal processing.

For details and restrictions, see "Signal Path" on page 73.

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
The narrowband signal path is used.

ON | 1
The wideband signal path is used.

*RST: 0

Example: `SENS:IQ:WBAN ON`

Manual operation: See "Signal Path" on page 73

5.6.6 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 84

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 81

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 83

CONFigure:GENerator:LEVel:DUTLimit <Value>

This command defines the output power RMS level of the generator.

Parameters:

<Value> <numeric value>

Default unit: dB

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 83

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 FSW).

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 83

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 83

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 83

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 83

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 FSW is changed
`CONF:GEN:FREQ:CENT:SYNC ON;*WAI`

Manual operation: See "Attach to Analyzer Frequency" on page 83

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 82

CONFigure:GENerator:CONNection:CSTate?

Queries the state of the connected signal generator.

Return values:

<ConnectionState> **UNKNown**
no signal generator connected
CONNected
connection established
NCONnected
connection could not be established, possibly due to an incompatible instrument or invalid IP address

Example: `CONFigure:GENerator:CONNection:CSTate?`

Usage: Query only

Manual operation: See "IP Address" on page 82

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 82

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 84

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 82

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 82

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 82

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 84

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 84

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 83

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 83

CONFigure:GENerator:SETTings:UPDate

This command updates the generator settings as defined within the R&S FSW-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 83

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 83

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

CONFigure:GENerator:RELay:READ?

Provides functionality to read the answer if the command that was sent to the signal generator using `CONFigure:GENerator:RELay:WRITe` on page 254 contained a "?".

Return values:
<Response>

Example: `CONF:GEN:REL:READ?`

Usage: Query only
Asynchronous command

CONFigure:GENerator:RELay:WRITe <Command>

Provides functionality to configure the signal generator directly through the FSW-K18 application. It resends the string parameter as a SCPI command to the connected signal generator.

If the command contains a "?", use `CONFigure:GENerator:RELay:READ?` on page 254 to read the answer.

Setting parameters:
<Command>

Example: `CONF:GEN:REL:WRIT "generator command"`

Usage: Setting only
Asynchronous command

5.6.7 Configuring the data capture

[SENSe<ip>:]BANDwidth[:RESolution] <Bandwidth>

This command defines the resolution bandwidth applied to spectrum measurements.

Prerequisites for this command

- Turn off automatic selection of RBW (`[SENSe<ip>:]BANDwidth[:RESolution]:AUTO`).

Suffix:

<ip>	1..n

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 136

[SENSe<ip>:]BANDwidth[:RESolution]:AUTO <State>

This command turns automatic selection of the resolution bandwidth (RBW) for spectrum measurements on and off.

Suffix:

<ip>	1..n

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 136

[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 98

[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 98

[SENSe:]SWEep:LENGth <Samples>

This command defines the capture length.

Prerequisites for this command

- Turn off automatic selection of the capture time (`[SENSe<ip>:]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 98

[SENSe<ip>:]SWEep:TIME <Time>

This command defines the capture time.

Prerequisites for this command

- Turn off automatic selection of the capture time (`[SENSe<ip>:]SWEep:TIME:AUTO`).

Effects of this command

- Changing the capture time automatically adjusts the capture length.

Suffix:
<ip> 1..n

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 98

[SENSe<ip>:]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.

Suffix:
<ip> 1..n

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 98

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 97

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 97

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 96

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 FSW.

The command is available when you turn on `TRACe:IQ:WBANd[:STATe]`.

Parameters:
<Bandwidth>

80MHZ
Restricts the bandwidth to 80 MHz.
(The wideband signal path is not used in that case. `TRACe:IQ:WBANd[:STATe]` is turned off.)

160MHZ | 320MHZ | 500MHZ
Restricts the bandwidth to the corresponding value, even if you have installed a higher bandwidth extension.

Default unit: Hz

Example: //Restrict the bandwidth to 160 MHz
`TRAC:IQ:WBAN ON`
`TRAC:IQ:WBAN:MBW 160MHZ`

Manual operation: See "Maximum bandwidth" on page 97

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 FSW.

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 80 MHz.

OFF | 0
Turns off the wideband signal path. The largest available bandwidth is 80 MHz.

Example: //Turn off the wideband signal path
`TRAC:IQ:WBAN OFF`

Manual operation: See "Maximum bandwidth" on page 97

5.6.8 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 "I/Q Averaging Sweep Count" on page 125

[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`

[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

[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 State" on page 99

[SENSe:]SWEep:STATistics:COUNt <Count>

Sets and queries the sweep statistics count.

Parameters:
<Count> numeric value

Example: `SENS:SWE:STAT:COUNt 10`

Manual operation: See "Statistics Count" on page 100

[SENSe:]SWEep:STATistics:CONTinuous[:STATe] <State>

Sets and queries the continuous statistics setting.

Parameters:
<State> ON | OFF

Example: `[SENSe]:SWEep:STATistics:CONT ON`

Manual operation: See "Continuous Statistics" on page 99

[SENSe:]SWEep:STATistics:MODE <State>

Sets and queries the statistics mode.

Parameters:
<State> IQAVeraging | TRACe

Example: `[SENSe]:SWEep:STATistics:MODE TRACe`

Manual operation: See "Statistics Mode" on page 99

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 32
See "Select Result Rng" on page 100

5.6.9 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 103

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 263
- `CONFigure:ESTimation:STOP` on page 263

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 103

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 103

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 102

CONFigure:SYNC:SECond:STAT <State>

This command activates an additional synchronization algorithm (operating in frequency domain).

Parameters:

<State> ON | OFF | 1 | 0

*RST: 0

Example: //Turn on additional synchronization algorithm.
`:CONF:SYNC:SEC:STAT ON`

Manual operation: See "2nd Stage Synchronization" on page 103

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 102

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 101

FETCh[:SYNC]?

This command queries if synchronisation between reference and measured signal has been successful.

The command is available when you have turned on `CONFigure:SYNC:STAT`.

Return values:

<State> **ON | 1**
Synchronisation has been successful.
OFF | 0
Synchronisation has not been successful.

Example: `FETC?`
would return, e.g.
`0`

Usage: Query only

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

5.6.10 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 104

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 266
- CONFigure:EVALuation:STOP on page 267

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

Manual operation: See "Defining the evaluation range" on page 104

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 104

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 104

5.6.11 Estimating and compensating signal errors

5.6.11.1 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
```

Manual operation: See "Amplitude Droop" on page 106

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`

Manual operation: See "Frequency Error" on page 106

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`

Manual operation: See "I/Q Imbalance" on page 106

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`

Manual operation: See "I/Q Offset" on page 106

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`

Manual operation: See "Sample Error Rate" on page 106

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`

Manual operation: See "Amplitude Droop" on page 106

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`

Manual operation: See "Frequency Error" on page 106

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`

Manual operation: See "I/Q Imbalance" on page 106

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`

Manual operation: See "I/Q Offset" on page 106

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`

Manual operation: See "Sample Error Rate" on page 106

5.6.11.2 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 107

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 107
See "Equalizer Filter Length For Training" on page 125

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 107

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 107

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 107

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 107

MMEMory:STORe<n>: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`).

Suffix:

<n> 1..n

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 107

5.6.12 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 109

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 109

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 110

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 110

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 110

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 109

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 109

5.6.13 Applying digital predistortion

5.6.13.1 Polynomial DPD

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 114

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 114

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 114

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 277

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 113

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 112

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 114

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 113

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 "Polynomial DPD Power / Linearity Tradeoff" on page 113

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 280
  and
- CONFigure:DPD:AMXM[:STATe] on page 277

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

MMEMory:STORe<n>: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`.

Suffix:
<n> 1..n

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

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

FETCh:DPD:WAVeform:PATH?

Queries the path of the Polynomial DPD waveform.

Return values:
<FileName>

Example:
```
FETCh:DPD:WAV:PATH?
```

Usage: Query only

5.6.13.2 Direct DPD

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 286).
- Initiate a DPD sequence (`CONFigure:DDPD:STARt` on page 286).

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 "Automated direct DPD sequence" on page 116

CONFigure:DDPD:APPLy:WRAP[:STATe] <State>

Smoothes start- and tail-samples down to "0" in order to avoid phase discontinuities when the file is cyclically played from a signal source.

Parameters:
<State> ON | OFF | 1 | 0
*RST: 1

Example: `CONFigure:DDPD:APPLy:WRAP ON`

Manual operation: See "Automated direct DPD sequence" on page 116
See "Manual direct DPD sequence" on page 117

CONFigure:DDPD:CONTinue

Ccontinues direct DPD in manual mode.

Example: `CONFigure:DDPD:CONTinue`

Usage: Event

Manual operation: See "Manual direct DPD sequence" on page 117

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 "Automated direct DPD sequence" on page 116
See "Manual direct DPD sequence" on page 117

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

Manual operation: See "Manual direct DPD sequence" on page 117

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: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 "Automated direct DPD sequence" on page 116

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 "Automated direct DPD sequence" on page 116
See "Manual direct DPD sequence" on page 117

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 286).

Example: //Initiate direct DPD sequence
`CONF:DDPD:STAR`

Usage: Event

Manual operation: See "Automated direct DPD sequence" on page 116
See "Manual direct DPD sequence" on page 117

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:RMS[:CURRent]?

Returns the current RMS power level in manual direct DPD mode.

Return values:

<Level> <numeric value>

Example: `CONFigure:DDPD:RMS:CURRent`

Usage: Query only

Manual operation: See "Manual direct DPD sequence" on page 117

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 "Direct DPD Power / Linearity Tradeoff" on page 118

CONFigure:DDPD:WINDow<n>:RESult <Result>

Configures the result type of the DDPD Results result display.

Suffix:

<n> Window

Parameters:

<Result>

EVM
Error Vector Magnitude

ACL1
AClrLower1

ACU1
AClrUpper1

ACB1
AClrBalanced1

Example: `CONFigure:DDPD:WIND1:RESult EVM`

Manual operation: See "DDPD Results (FSW-K18D)" on page 18
See "Direct DPD Result Type" on page 154

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:DDPD:WAVeform:PATH?

Queries the path of the Direct DPD waveform.

Return values:
<FileName>

Example: `FETCh:DDPD:WAVE:PATH?`

Usage: Query only

Manual operation: See "Manual direct DPD sequence" on page 117

MMEMory:STORe<n>: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 286).
- Run a DPD sequence (`CONFigure:DDPD:STARt` on page 286).

Suffix:
<n> 1..n

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 "Automated direct DPD sequence" on page 116

5.6.13.3 Generalized memory polynomial

CONFigure:GMP:LAG:ORDer:MEMory <Level>

Sets the lagging cross-terms memory order of the generalized memory polynomial.

Parameters:

<Level> *RST: 5

Example: //Set lagging cross-terms memory order
`CONF:GMP:LAG:ORD:MEM 5`

CONFigure:GMP:LAG:ORDer:POLYnomial <Level>

Sets the lagging cross-terms polynomial order of the generalized memory polynomial.

Parameters:

<Level> *RST: 5

Example: //Set lagging cross-terms polynomial order
`CONF:GMP:LAG:ORD:POLY 5`

CONFigure:GMP:LAG:ORDer:XTERm <Level>

Sets the lagging cross-term order of the generalized memory polynomial.

Parameters:

<Level> *RST: 0

Example: //Set lagging cross-terms order
`CONF:GMP:LAG:ORD:XTER 0`

CONFigure:GMP:LEAD:ORDer:MEMory <Level>

Sets the leading cross-terms memory order of the generalized memory polynomial.

Parameters:

<Level> *RST: 5

Example: //Set leading cross-terms memory order
`CONF:GMP:LEAD:ORD:MEM 5`

CONFigure:GMP:LEAD:ORDer:POLYnomial <Level>

Sets the leading cross-terms polynomial order of the generalized memory polynomial.

Parameters:

<Level> *RST: 5

Example: //Set leading cross-terms polynomial order
`CONF:GMP:LEAD:ORD:POLY 5`

CONFigure:GMP:LEAD:ORDer:XTERm <Level>

Sets the leading cross-term order of the generalized memory polynomial.

Parameters:

<Level> *RST: 0

Example: //Set leading cross-terms order
`CONF:GMP:LEAD:ORD:XTER 0`

CONFigure:MDPD:COEFficient:SELect <Type>

Selects the source from which the coefficients are to be used.

Parameters:

<Type> FILE | MODeling

FILE
Uses the coefficients from a `csv` file.

MODeling
Uses the coefficients generated by the modelling.

*RST: FILE

Example: //Select source
`CONFigure:MDPD:COEFficient:SELect FILE`

CONFigure:MDPD:APPLy:MODel <Channel>

Selects the waveform to which the model should be applied.

Info: First select the source to execute this command: `CONFigure:MDPD:COEFficient:SELect` on page 290.

Parameters:

<Channel>

Manual operation: See "Running a Memory Polynomial DPD sequence" on page 121

CONFigure:MDPD[:STATe] <State>

Switches the memory polynomial state on and off.

Parameters:
<State> **ON | 1**
OFF | 0
*RST: OFF

Example: `CONFigure:MDPD:STATe ON`

Manual operation: See "Running a Memory Polynomial DPD sequence" on page 121

CONFigure:MDPD:ITERation <Level>

Configures the iteration step for memory polynomial DPD.

Parameters:
<Level> <numeric value>

Example: `CONFigure:MDPD:ITERation 4`

Manual operation: See "Running a Memory Polynomial DPD sequence" on page 121

CALCulate:MDPD:MODel

Calculates the memory polynomial model.

Example: `CALCulate:MDPD:MODel`

Usage: Event

Manual operation: See "Running a Memory Polynomial DPD sequence" on page 121

CONFigure:MDPD:ORDer:POLYnomial <Order>

Sets the polynomial order for memory polynomial DPD as a string.

Parameters:
<Order>

Example: `CONFigure:MDPD:ORDer:POLY "1-7;9;11"`

Manual operation: See "Running a Memory Polynomial DPD sequence" on page 121

CONFigure:MDPD:ORDer:MEMory <Order>

Sets the memory order for memory polynomial DPD as a string.

Parameters:
<Order>

Example: `CONFigure:MDPD:ORDer:MEMory "1-7;9;11"`

Manual operation: See "Running a Memory Polynomial DPD sequence" on page 121

CONFigure:MDPD:RMS[:CURRent]?

Returns the current RMS power level of the memory polynomial waveform.

Return values:
<Level> <numeric value>

Example: `CONFigure:MDPD:RMS:CURRent`

Usage: Query only

Manual operation: See "Manual direct DPD sequence" on page 117

CONFigure:MDPD:WAVeform:SELect <Type>

Selects the type of DPD waveform to be used.

Parameters:
<Type>

DDPD
Uses a direct DPD waveform.

MDPD
Uses a memory polynomial DPD waveform.

REF
Uses the reference signal.

Example: `CONFigure:MDPD:WAVeform:SELect MDPD`

Manual operation: See "Running a Memory Polynomial DPD sequence" on page 121

CONFigure:MDPD:WAVeform:UPDate

Sends the memory polynomial waveform to the signal generator.

Example: `CONFigure:MDPD:WAVeform:UPDate`

Usage: Event

Manual operation: See "Running a Memory Polynomial DPD sequence" on page 121

FETCh:MDPD:COEFficients?

Fetches the MDPD coefficient values.

Example: `FETCh:MDPD:COEFficients?`

Usage: Query only

Manual operation: See "Memory DPD Coefficients" on page 23

FETCh:MDPD:WAVeform:PATH?

Queries the path of the Memory Polynomial DPD waveform.

Return values:
<FileName>

Example: `FETCh:MDPD:WAV:PATH?`

Usage: Query only

MMEMory:STORe:MDPD:COEFficient <FileName>

Exports the memory DPD coefficients in a file in `.csv` format.

Setting parameters:
<FileName>

Example: `MMEMory:STORe:MDPD:COEFficient 'C:\MemoryPolyCoeff.csv'`

Usage: Setting only

Manual operation: See "Running a Memory Polynomial DPD sequence" on page 121

MMEMory:STORe:MDPD:WAVeform <FileName>

Saves the memory polynomial waveform at a user selected path.

Only available when generator control is OFF.

Setting parameters:
<FileName>

Example: `MMEMory:STORe:MDPD:WAVeform 'C:\MemoryPoly.wv'`

Usage: Setting only

Manual operation: See "Running a Memory Polynomial DPD sequence" on page 121

5.6.13.4 Hammerstein model

CONFigure:HAMMerstein[:STATe] <State>

Switches Hammerstein mode on and off.

Parameters:

<State> **ON | 1**

OFF | 0

Example: `CONF:HAMM ON`

Manual operation: See "Running a Hammerstein model sequence" on page 124

CONFigure:HAMMerstein:ITERation <Level>

Configures the iteration step for the Hammerstein model.

Parameters:

<Level> <numeric value>

Example: `CONFigure:HAMM:ITERation 4`

Manual operation: See "Running a Hammerstein model sequence" on page 124

CONFigure:HAMMerstein:ORDer:POLYnomial <Order>

Sets the polynomial order for the Hammerstein model as a string.

Parameters:

<Order>

Example: `CONFigure:HAMM:ORDer:POLY "1-7;9;11"`

Manual operation: See "Running a Hammerstein model sequence" on page 124

CONFigure:HAMMerstein:ORDer:MEMory <Order>

Sets the memory order for the Hammerstein model as a string.

Parameters:

<Order>

Example: `CONFigure:HAMMerstein:ORDer:MEMory "1-7;9;11"`

Manual operation: See "Running a Hammerstein model sequence" on page 124

CONFigure:HAMMerstein:MUPGenerator

Starts the DSP and updates the generator.

Example: `CONFigure:HAMMerstein:MUPGenerator`

Usage: Event

Manual operation: See "Running a Hammerstein model sequence" on page 124

CONFigure:HAMMerstein:GENWaveform[:SELect] <GeneratorWaveform>

Switches the generator waveform between reference and direct DPD.

Parameters:

<GeneratorWaveform> REFerence | DDPD

REFerence
Reference waveform

DDPD
DDPD waveform

Example: `CONFigure:Hammerstein:GENWaveform REF`

Manual operation: See "Running a Hammerstein model sequence" on page 124

CONFigure:HAMMerstein:NONLinearity[:STATe] <State>

Switches the non-linearity (SMx-K541) on and off.

Parameters:

<State> **ON | 1**

OFF | 0

Example: `CONFigure:Hammerstein:NONLinearity ON`

Manual operation: See "Running a Hammerstein model sequence" on page 124

CONFigure:HAMMerstein:FILTer[:STATe] <State>

Switches the filter (SMx-K544) on and off.

Parameters:

<State> **ON | 1**

OFF | 0

Example: `CONFigure:Hammerstein:FILTer ON`

Manual operation: See "Running a Hammerstein model sequence" on page 124

5.6.14 Detailed MSE

CALCulate:MSERror?

Calculates the detailed MSE and returns the result.

Example: `CALC:MSER?`

Usage: Query only

Manual operation: See "Calculate Detailed MSE" on page 126

CALCulate:MSERror:CONFigure:EQUalizer:FILTer:LENGth <Length>

Parameters:
<Length>

CALCulate:MSERror:IQAVg:COUNt <Count>

Parameters:
<Count>

5.6.15 Trigger

TRIGger[:SEQuence]:CTPis <State>

The trigger event is set to the actual trigger point within the sample (TPIS). See also `TRACe:IQ:TPISample?` on page 297.

This setting is available for all trigger sources except "Free Run".

This setting only becomes effective for the next new measurement.

TPIS correction is not supported if user frequency response correction (R&S FSW-K544) is active. If you activate frequency response correction, TPIS correction is automatically deactivated in all channels.

For details see frequency response correction in the FSW user manual.

Parameters:

<State> ON | OFF | 0 | 1

OFF | 0
Switches the function off

ON | 1
Switches the function on

*RST: 0

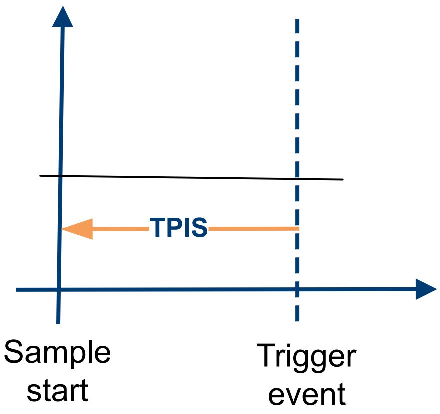
Example:

```
//    Query the current trigger offset within the sample
TRAC:IQ:TPIS?
//Result: 0.000002
//Correct the trigger event
TRIG:CTP ON
//Start a new measurement
INIT:IMM; *WAI
//Query the trigger offset within the sample again
TRAC:IQ:TPIS?
//Result: 0.000000
```

Manual operation: See "Correct TPIS" on page 95

TRACe:IQ:TPISample?

Queries the time offset from the sample start to the trigger event (trigger point in sample = TPIS). Since the FSW usually samples with a much higher sample rate than the specific application actually requires, the trigger point determined internally is much more precise than the one determined from the (downsampled) data in the application. Thus, the TPIS indicates the offset from the sample start to the actual trigger event.



This value can only be determined in triggered measurements using external or IFPower triggers, otherwise the value is 0.

Is not available if the "Digital Baseband" interface (FSW-B17) is active.

Return values:

<TPIS> numeric value

Default unit: s

Example: `TRAC:IQ:TPIS?`

Result for a sample rate of 1 MHz: between 0 and 1/1 MHz, i.e. between 0 and 1 μs (the duration of 1 sample).

Usage: Query only

5.6.16 Configuring envelope tracking

CONFigure:PAE:ICHannel:MULTiplier <Multiplier>

This command defines a multiplier to take into account various effects resulting from the measurement equipment connected to the I channel.

Parameters:

<Multiplier> <numeric value>

Example: //Defines a multiplier of 0.75

```
CONF:PAE:ICH:MULT 0.75
```

CONFigure:PAE:ICHannel:OFFSet <Offset>

This command defines an offset for the I channel.

Parameters:

<Offset> <numeric value>

Default unit: No unit

Example: //Define an offset of 1

```
CONF:PAE:ICH:EOFF 1
```

CONFigure:PAE:ICHannel:RESistor <Resistance>

This command defines the characteristics of the shunt resistor used in the test setup.

Parameters:

<Resistance> <numeric value>

Resistance in Ohm.

Example: //Defines a resistance of 1.5 Ω

```
CONF:PAE:ICH:RES 1.5
```

CONFigure:PAE:PCONsumption[:PARameter]:A <Value>

Parameters:

<Value>

CONFigure:PAE:PCONsumption[:PARameter]:B <Value>

Parameters:
<Value>

CONFigure:PAE:QCHannel:MULTiplier <Multiplier>

This command defines a multiplier to take into account various effects resulting from the measurement equipment connected to the Q channel.

Parameters:
<Multiplier> <numeric value>

Example: //Define a multiplier of 1.2
```
CONF:PAE:QCH:MULT 1.2
```

CONFigure:PAE:QCHannel:OFFSet <Offset>

This command defines an offset for the Q channel.

Parameters:
<Offset> <numeric value>
Default unit: No unit

Example: //Defines an offset of 1
```
CONF:PAE:QCH:OFFS 1
```

5.6.17 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 13

[SENSe:]POWer:ACHannel:ACPairs <ChannelPairs>

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 129

[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 129

[SENSe:]POWer:ACHannel:BANDwidth:ACHannel <Bandwidth>

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 FSW 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 130

[SENSe:]POWer:ACHannel:BANDwidth:ALTernate<ch> <Bandwidth>

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 130

[SENSe:]POWer:ACHannel:BANDwidth[:CHANnel<ch>] <Bandwidth>

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 130

[SENSe:]POWer:ACHannel:FILTer:ALPHa:ACHannel <Alpha>

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 131

[SENSe:]POWer:ACHannel:FILTer:ALPHa:ALTernate<ch> <Alpha>

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 131

[SENSe:]POWer:ACHannel:FILTer:ALPHa:CHANnel<ch> <Alpha>

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 131

[SENSe:]POWer:ACHannel:FILTer[:STATe]:ACHannel <State>

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 131

[SENSe:]POWer:ACHannel:FILTer[:STATe]:ALTernate<ch> <State>

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 131

[SENSe:]POWer:ACHannel:FILTer[:STATe]:CHANnel<ch> <State>

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 131

[SENSe:]POWer:ACHannel:REFerence:TXCHannel:AUTO <RefChannel>

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 130

[SENSe:]POWer:ACHannel:REFerence:TXCHannel:MANual <ChannelNumber>

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 130

[SENSe:]POWer:ACHannel:SPACing:CHANnel<ch> <Spacing>

Defines the distance between transmission channels.

If you set the channel spacing for a transmission channel, the FSW 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 130

[SENSe:]POWer:ACHannel:SPACing[:ACHannel] <Spacing>

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 130

[SENSe:]POWer:ACHannel:SPACing:ALTernate<ch> <Spacing>

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 130

[SENSe:]POWer:ACHannel:TXCHannel:COUNt <Number>

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 129

5.6.18 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 128

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 128

5.6.19 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 134

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 134

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 133

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.

5.6.20 Configuring power servoing

Note that as long as you have selected automatic definition of the data acquisition values in the Power Servoing measurement, the application takes the same values as those that have been defined for the basic amplifier measurement (this is also the case when the basic data acquisition parameters are selected automatically).

If you want to define values specific to the Power Servoing measurement, you have to turn off automatic definition of the data acquisition value. If you turn automatic definition back on, the values are synchronized to those of the basic amplifier measurement again.

[SENSe:]PSERvoing:STATe <State>

Sets and queries the power servoing state.

Parameters:

<State> ON | OFF | 1 | 0

Example: `PSER:STAT ON`

Manual operation: See "Power Servoing sequence" on page 135

[SENSe:]PSERvoing:TARGet:PARameter <Target>

Sets the power servoing target parameter.

Parameters:

<Target> POUT | EVM | LADJ | UADJ | LALT | UALT

POUT
Power Out

EVM
EVM

LADJ
ACLR Adjacent Lower

UADJ
ACLR Adjacent Upper

LALT
ACLR Alternate Lower

UALT
ACLR Alternate Upper

Example: `PSER:TARG:PAR POUT`

Manual operation: See "Power Servoing sequence" on page 135

[SENSe:]PSERvoing:TARGet:VALue <TargetValue>

Sets and queries the power servoing target value. The unit depends on the selected target parameter.

Parameters:
<TargetValue> <numeric value>

Example: `PSER:TARG:VAL 3`

Manual operation: See "Power Servoing sequence" on page 135

[SENSe:]PSERvoing:TARGet:TOLerance <TargetValue>

Sets and queries the power servoing target tolerance. The unit depends on the selected target parameter.

Parameters:
<TargetValue> <numeric value>

Example: `PSER:TARG:TOL 0.2`

Manual operation: See "Power Servoing sequence" on page 135

[SENSe:]PSERvoing:MAX:ITERation <MaxIterations>

Sets and queries the maximum number of iteratons during the power servoing sequence.

Parameters:
<MaxIterations> <numeric value>

Example: PSER:MAX:ITER 5

Manual operation: See "Power Servoing sequence" on page 135

[SENSe:]PSERvoing[:GLC] <GenLevelControl>

Selects if the generator level is modified using input power or digital attenuation.

Parameters:
<GenLevelControl> RFL | DATT

RFL
Input power

DATT
Digital attenuation

Example: `PSER:GLC RFL`

Manual operation: See "Power Servoing sequence" on page 135

[SENSe:]PSERvoing:INPut:STEP <InputPowerStep>

Defines the input power step size.

Parameters:
<InputPowerStep> <numeric value>

Manual operation: See "Power Servoing sequence" on page 135

[SENSe:]PSERvoing:STARt

Starts the power servoing sequence.

Example: `PSER:STAR`

Usage: Event

Manual operation: See "Power Servoing sequence" on page 135

FETCh:PSERvoing:OPERation:STATus?

Queries the status of the power servoing operation.

Return values:
<State> ON | OFF | 1 | 0

Example: `FETCh:PSERvoing:OPERation:STATus?`

Usage: Query only

Manual operation: See "Power Servoing sequence" on page 135

5.6.21 Frequency domain measurements

CONFigure:FDOMain:FFTLength <FFT Length>

Defines the number of frequency points determined by each FFT calculation. The more points are used, the higher the resolution in the spectrum becomes, but the longer the calculation takes.

Parameters:
<FFT Length> Range: 1k to 32k
*RST: 2k
Default unit: -

Example: `CONF:FDOM:FFTL 2048`

CONFigure:FDOMain:SQUelch <SquelchLevel>

For group delay results, defines a level threshold below which the group delay is set to 0. If the group delay does not exceed the threshold, it is ignored altogether.

Parameters:

<SquelchLevel>
Range: -200 to +200
Increment: 0.1
*RST: -150.0
Default unit: dBm

Example: `CONF:FDOM:SQU 20`

Manual operation: See "Use Squelch for Group Delay" on page 137

CONFigure:FDOMain:WSETtings:AUTO <State>

Sets the the window settings mode on or off.

Parameters:

<State> ON | OFF | 1 | 0

ON | 1
Sets the window settings mode on.

OFF | 0
Sets the window settings mode off.

Example: //Activate window settings mode.
`CONF:FDOM:WSET:AUTO ON`

Manual operation: See "Window Settings" on page 136

CONFigure:FDOMain:WFUNction <Method>

Defines the FFT window type.

Parameters:

<Method> FLATtop | GAUSsian | RECTangular | P5 | BLACkharris
*RST: FLATtop

Example: `CONF:FDOM:WFUN GIAN`

Manual operation: See "Window Function" on page 137

CONFigure:FDOMain:WLFRatio <WLength Ratio>

Defines the window length as a percentage of the FFT length (see `CONFigure:FDOMain:FFTLength` on page 312).

Parameters:

<WLength Ratio>
Range: 0.1 to 100
Increment: 0.1
*RST: 25
Default unit: percent

Example: `CONF:FDOM:WLFR 25`

CONFigure:FDOMain:WOVerlap <Window Overlap>

Defines the part of a single FFT window that is re-calculated by the next FFT calculation when using multiple FFT windows.

Parameters:

<Window Overlap>
Range: 0 to 99.9
Increment: 0.1
*RST: 25
Default unit: percent

Example: `CONF:FDOM:WOV 25`

Manual operation: See "Window Overlap" on page 137

5.6.22 Adjusting settings automatically

The commands required to adjust settings automatically in a remote environment are described here. The tasks for manual operation are described in Chapter 3.21, "Adjusting settings automatically", on page 137.

[SENSe:]ADJust:LEVel

Initiates a single (internal) measurement that evaluates and sets the ideal reference level for the current input data and measurement settings. Thus, the settings of the RF attenuation and the reference level are optimized for the signal level. The FSW is not overloaded and the dynamic range is not limited by an S/N ratio that is too small.

Example: `ADJ:LEV`

Manual operation: See "Setting the Reference Level Automatically (Auto Level)" on page 138

[SENSe:]AUTO:FAST

Activates the "Auto Fast" function to improve the measurement speed.

Example: //Activate "Auto Fast" function.
`AUTO:FAST`

Usage: Event

Manual operation: See "Optimize the evaluation range settings for speed (Auto Fast)" on page 138

[SENSe:]AUTO:ROBust

Activates the "Auto Robust" function to add robustness in the measurement.

Example: //Activate "Auto Robust" function.
`AUTO:ROB`

Usage: Event

Manual operation: See "Optimize the evaluation range settings for robustness (Auto Robust)" on page 138

5.7 Analyzing results

5.7.1 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 FSW 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 FSW 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 141

FORMat:DEXPort:DSEParator <Separator>

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 143

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 143

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>` and the trace id, `<Trace>`, are 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 file.
```
FORM:DEXP:TRAC ALL
MMEM:STOR:TRAC 0,'C:\TraceResults'
```
//Export all traces in window 2 to the specified file.
```
FORM:DEXP:TRAC SING
MMEM:STOR2:TRAC 0,'C:\TraceResults'
```
//Export the second trace in window 2 to the specified file.
```
MMEM:STOR2:TRAC 2,'C:\TraceResults'
```

Usage: Setting only

Manual operation: See "Selecting data to export" on page 143
See "Export Trace" on page 144

[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 144

[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 144

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

Info: The parameter "REFerence" also refers to "IdealLine".

Example: `DISP:WIND:TRAC:RES MEAS`

Manual operation: See "Result Type" on page 142

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

Manual operation: See "Predefined Trace Settings - Quick Config" on page 142

[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 142

TRACe:IQ:DATA?

Sweeps and transfers raw I/Q data.

Example: `TRACe1:IQ:DATA?`

Usage: Query only

TRACe:IQ:DATA:FORMat <Slope>

Defines the I/Q data format.

Parameters:

<Slope> COMPatible | IQBLock | IQPair

Example: `TRACe1:IQ:DATA:FORMat IQP`

TRACe:IQ:DDPD[:DATA]?

Queries the I/Q values of the current direct DPD iteration (only for unencrypted files).

Example: `TRACe1:IQ:DDPD[:DATA]?`

Usage: Query only

Manual operation: See "Automated direct DPD sequence" on page 116
See "Manual direct DPD sequence" on page 117

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

TRACe:IQ:TPIS?

Return values:

<time> Default unit: HZ

Usage: Query only

5.7.2 Using markers

5.7.2.1 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 146

DISPlay[:WINDow<n>]:MINFo[:STATe] <State>

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 145

DISPlay[:WINDow<n>]:MTABle <DisplayMode>

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 145

5.7.2.2 Configuring individual markers

CALCulate<n>:DELTamarker<m>:AOFF

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>

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 147

CALCulate<n>:DELTamarker<ms>:LINK:TO:MARKer<md> <State>

Links the delta source marker <ms> to any active destination marker <md> (normal or delta marker).

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 147

CALCulate<n>:DELTamarker<m>:MREFerence <Reference>

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 147

CALCulate<n>:DELTamarker<m>[:STATe] <State>

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 147
See "Marker Type" on page 147

CALCulate<n>:DELTamarker<m>:TRACe <Trace>

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>

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 147

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

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 148

CALCulate<n>:MARKer<ms>:LINK:TO:MARKer<md> <State>

Links the normal source marker <ms> to any active destination marker <md> (normal or delta marker).

If you change the horizontal position of marker <md>, marker <ms> 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 147

CALCulate<n>:MARKer<m>[:STATe] <State>

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 147
See "Marker Type" on page 147

CALCulate<n>:MARKer<m>:TRACe <Trace>

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 148

CALCulate<n>:MARKer<m>:X <Position>

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 147

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

5.7.2.3 Positioning markers

CALCulate<n>:DELTamarker<m>:MAXimum:LEFT

Moves a delta marker to the next positive peak 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 149

CALCulate<n>:DELTamarker<m>:MAXimum:NEXT

Moves a marker to the next positive peak value.

Suffix:

<n> 1..n
Window

<m> 1..n
Marker

Manual operation: See "Search Next Peak" on page 149

CALCulate<n>:DELTamarker<m>:MAXimum[:PEAK]

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 148

CALCulate<n>:DELTamarker<m>:MAXimum:RIGHt

Moves a delta marker to the next positive peak value on the trace.

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 149

CALCulate<n>:DELTamarker<m>:MINimum:LEFT

Moves a delta marker to the next minimum peak 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 149

CALCulate<n>:DELTamarker<m>:MINimum:NEXT

Moves a marker to the next minimum peak value.

Suffix:

<n> Window

<m> Marker

Manual operation: See "Search Next Minimum" on page 149

CALCulate<n>:DELTamarker<m>:MINimum[:PEAK]

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 149

CALCulate<n>:DELTamarker<m>:MINimum:RIGHt

Moves a delta marker to the next minimum peak 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 149

CALCulate<n>:MARKer<m>:MAXimum:LEFT

Moves a marker to the next positive 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 149

CALCulate<n>:MARKer<m>:MAXimum:NEXT

Moves a marker to the next positive peak.

Suffix:

<n> Window

<m> Marker

Manual operation: See "Search Next Peak" on page 149

CALCulate<n>:MARKer<m>:MAXimum[:PEAK]

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 148

CALCulate<n>:MARKer<m>:MAXimum:RIGHt

Moves a marker to the next positive 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 149

CALCulate<n>:MARKer<m>:MINimum:LEFT

Moves a marker to the next minimum peak 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 149

CALCulate<n>:MARKer<m>:MINimum:NEXT

Moves a marker to the next minimum peak value.

Suffix:

<n> Window

<m> Marker

Manual operation: See "Search Next Minimum" on page 149

CALCulate<n>:MARKer<m>:MINimum[:PEAK]

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 149

CALCulate<n>:MARKer<m>:MINimum:RIGHt

Moves a marker to the next minimum peak 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 149

5.7.3 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
APAE	Average PAE
BBIVoltage	Baseband I input voltage
BBPower	Baseband power
BBQVoltage	Baseband Q input voltage
CFIN	Crest factor in
CFOU	Crest factor out
FERRor	Frequency error
GAIN	Gain
GIMBalance	Gain Imbalance
ICC	Current
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
PCPA	PC based average PAE
PERRor	Phase error
PINPut	Power in
PMWidth	PM curve width
POUTput	Power out
QERRor	Quadrature error
REVM	"Raw EVM"
RMEVm	Raw model EVM

SCPI parameter	Result
SRERror	Sample rate error
VCC	Voltage

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

DISPlay[:WINDow<n>]:TABLe:ITEM:VCURrent:ALL <State>

This command adds and removes all voltage / current results from the result summary.

Prerequisites for this command

- Select a baseband measurement.

Suffix:
<n> Window

Setting parameters:
<State> ON | OFF | 1 | 0

Example: //Displays all voltage and current results
`DISP:TABL:ITEM:VCUR:ACC:ALL ON`

Usage: Setting only

5.7.4 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 332 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

DISPlay[:WINDow<n>]:STABle:ITEM:VCURrent:ALL <State>

This command adds and removes all voltage / current results from the statistics table.

Prerequisites for this command

- Select a baseband measurement.

Suffix:
<n> Window

Setting parameters:
<State> ON | OFF | 1 | 0

Example: //Displays all voltage and current results
`DISP:STAB:ITEM:VCUR:ACC:ALL ON`

Usage: Setting only

5.7.5 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 "AM/PM Definition" on page 152

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 "Reference for AM/PM, EVM and Gain Compression" on page 154

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 "Unit" on page 152

CONFigure:FRSPan <Time>

Sets or queries the the frequency response span for FSW-K18F result displays.

Parameters:
<Time> Range: 1 Hz to 100 GHz
Default unit: HZ

Example:
```
CONF:FRSP:AUTO OFF
CONF:FRSP 2000
```

Manual operation: See "Frequency Response" on page 152

CONFigure:FRSPan:AUTO <State>

Defines how the span is determined that the frequency response is applied to for FSW-K18F result displays.

Parameters:
<State> ON | OFF | 0 | 1

OFF | 0
Defines the span manually using `CONFigure:FRSPan` on page 338.

ON | 1
Defines the span automatically according to the calculated OBW of the reference file.

*RST: 0

Manual operation: See "Frequency Response" on page 152

CONFigure:POWer:UNIT <Result>

Switches the unit for power results from dBm (default) to Watts.

Parameters:
<Result> DBM | WATT

Manual operation: See "Power Unit" on page 152

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 "Parameter sweep diagram" on page 153

ACBM	Balanced ACLR Magnitude
ACB1	ACLR Adj 1 Balanced
ACB2	ACLR Alt 1 Balanced
ACB3	ACLR Alt 2 Balanced
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
ICC	Current (I_cc)
PAE	PAE
PMWidth	"AM/PM" Curve Width
POUT	Power Out
P1DB	Compression Point 1 dB
P2DB	Compression Point 2 dB
P3DB	Compression Point 3 dB
RMS	RMS Power
VCC	Voltage (V_cc)
VICC	Power (V_cc * I_cc)

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 "Duration" on page 153

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 "Position" on page 153

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 "Offset" on page 153

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 "Normalize to 1" on page 153

CONFigure:AMPM:CWIDth:REFerence <CurveWidthReference>

Sets and queries the curve width computation reference point

Parameters:

<CurveWidthReference> <numeric value>

Default unit: dB

Example:

```
CONF:AMPM:CWID:REF 3DB
```

Manual operation: See "AM/AM and AM/PM Curve Width Reference Point" on page 152

CONFigure:AMPM:CWIDth:REFerence:AUTO <State>

Sets and queries the curve width computation refrence point mode.

Parameters:

<State> **ON | 1**

Automatic mode

OFF | 0

Manual mode

Example:

```
CONFigure:AMPM:CWIDth:REFerence:AUTO ON
```

Manual operation: See "AM/AM and AM/PM Curve Width Reference Point" on page 152

5.7.6 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 155

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 155

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 155

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 155

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 157

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 157

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 157

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 157

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 157

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 157

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

5.8 Retrieving results

The following remote commands are required to retrieve the results from an amplifier measurement in a remote environment.

5.8.1 Retrieving graphical measurement results

FORMat[:DATA] <Format>[, <BitLength>]

Selects the data format that is used for transmission of trace data from the FSW to the controlling computer.

Note that the command has no effect for data that you send to the FSW. The FSW 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 can 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 14
See "AM/PM (Input Power)" on page 15
See "Channel Response Magnitude / Channel Response Phase / Group Delay (FSW-K18F)" on page 16
See "DDPD Results (FSW-K18D)" on page 18
See "EVM vs Power" on page 19
See "Error Vector Spectrum" on page 19
See "Gain Compression" on page 20
See "Gain Deviation vs Time" on page 21
See "Vcc vs Icc" on page 22
See "Magnitude Capture" on page 22
See "PAE vs Input Power / PAE vs Output Power" on page 23
See "PAE vs Time" on page 24
See "Parameter Sweep: Diagram" on page 26
See "Phase Deviation vs Time" on page 26
See "Power vs Time" on page 27
See "Raw EVM" on page 27
See "Spectrum FFT" on page 29
See "Time Domain" on page 30
See "Vcc vs Power / Icc vs Power" on page 32

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.

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

5.8.2 Retrieving numeric results

The following commands are required to retrieve the calculated numeric results.

5.8.2.1 Retrieving general numeric results

FETCh:TTF:CURRent[:RESult]?

This command queries the Trigger to Frame result as displayed in the channel bar.

Return values:

<Time> <numeric value>

Default unit: s

Example: `FETC:TTF:CURR?`
would return, e.g.
`0.00015700958`

Usage: Query only

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

5.8.2.2 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:ADRoop:MAXimum[:RESult]?
FETCh:MACCuracy:ADRoop:MINimum[:RESult]?
FETCh:MACCuracy:ADRoop:CURRent[:RESult]?

Return values:
<ADRoop>

Usage: Query only

Manual operation: See "Amplitude Droop" on page 34

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 "Frequency Error" on page 34

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 "Gain Imbalance" on page 34

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 "I/Q Imbalance" on page 34

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 "I/Q Offset" on page 35

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 "Magnitude Error" on page 35

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 "Phase Error" on page 35

FETCh:MACCuracy:POFFset[:RESult]?

Queries the absolute phase value between reference signal and measured signal.

Note that the absolute phase is not relevant for FSW-K18 measurements.

However, it can be used to track the absolute phase stability between generator and analyzer (including their local oscillators).

Return values:
<Phase Offset> Numeric value

Default unit: radian

Example: `FETC:MACC:POFF:RES?`
Returns the phase offset in radians.

Usage: Query only

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 "Quadrature Error" on page 36

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 "Raw EVM" on page 37

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 "Raw Model EVM" on page 37

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 "Sample Rate Error" on page 37

Retrieving power results

FETCh:AMAM:CWIDth:MAXimum[:RESult]?
FETCh:AMAM:CWIDth:MINimum[:RESult]?
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: V

Example: `FETC:AMAM:CWID:CURR?`
would return, e.g.
`0.69`

Usage: Query only

Manual operation: See "AM/AM Curve Width" on page 38

FETCh:AMPM:CWIDth:MAXimum[:RESult]?
FETCh:AMPM:CWIDth:MINimum[:RESult]?
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 "AM/PM Curve Width" on page 39

FETCh:AMPM:PEAK:CWIDth:MAXimum[:RESult]?
FETCh:AMPM:PEAK:CWIDth:MINimum[:RESult]?
FETCh:AMPM:PEAK:CWIDth:CURRent[:RESult]?

This command queries the "AM/PM" peak curve width as shown in the result summary.

Return values:
<CurveWidth> <numeric value>

Current "AM/PM" peak curve width.

Default unit: °

Example: `FETC:AMPM:PEAK:CWID:CURR:RES?`

Usage: Query only

Manual operation: See "AM/PM Curve Width (Pk-Pk)" on page 40

FETCh:AMAM:PEAK:CWIDth:MAXimum[:RESult]?
FETCh:AMAM:PEAK:CWIDth:MINimum[:RESult]?
FETCh:AMAM:PEAK:CWIDth:CURRent[:RESult]?

This command queries the "AM/AM" peak curve width as shown in the result summary.

Return values:

<CurveWidth> <numeric value>

Current "AM/AM" peak curve width.

Default unit: dB

Example: `FETC:AMAM:PEAK:CWID:CURR:RES?`

Usage: Query only

Manual operation: See "AM/AM Curve Width (Pk-Pk)" on page 39

FETCh:PC:CURRent[:RESult]?

Return values:

<Current>

Usage: Query only

FETCh:PCPA:CURRent[:RESult]?

Return values:

<Current>

Usage: Query only

FETCh:POWer:CFACtor:IN:MAXimum[:RESult]?
FETCh:POWer:CFACtor:IN:MINimum[:RESult]?
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 "Crest Factor In" on page 41

FETCh:POWer:CFACtor:OUT:MAXimum[:RESult]?
FETCh:POWer:CFACtor:OUT:MINimum[:RESult]?
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 "Crest Factor Out" on page 41

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 "Gain" on page 41

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 "Power In" on page 42

FETCh:POWer:OBW:MAXimum[:RESult]?
FETCh:POWer:OBW:MINimum[:RESult]?
FETCh:POWer:OBW:CURRent[:RESult]?
Return values:
<Level>

Usage: Query only

Manual operation: See "Occupied Bandwidth" on page 42

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 "Power Out" on page 42

FETCh:POWer:OUTPut:P1DB:MAXimum[:RESult]?
FETCh:POWer:OUTPut:P1DB:MINimum[:RESult]?
Return values:
<Level>

Usage: Query only

FETCh:POWer:OUTPut:P2DB:MAXimum[:RESult]?
FETCh:POWer:OUTPut:P2DB:MINimum[:RESult]?
Return values:
<Level>

Usage: Query only

FETCh:POWer:OUTPut:P3DB:MAXimum[:RESult]?
FETCh:POWer:OUTPut:P3DB:MINimum[:RESult]?

Return values:
<Level>

Usage: Query only

FETCh:POWer:P1DB:MAXimum[:RESult]?
FETCh:POWer:P1DB:MINimum[:RESult]?
FETCh:POWer:P1DB:CURRent[:RESult]?
FETCh:POWer:P2DB:MAXimum[:RESult]?
FETCh:POWer:P2DB:MINimum[:RESult]?
FETCh:POWer:P2DB:CURRent[:RESult]?
FETCh:POWer:P3DB:MAXimum[:RESult]?
FETCh:POWer:P3DB:MINimum[:RESult]?
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 "Compression Point (1 dB / 2 dB / 3 dB)" on page 40

FETCh:POWer:P1DB:OUT:MAXimum[:RESult]?
FETCh:POWer:P1DB:OUT:MINimum[:RESult]?
FETCh:POWer:P1DB:OUT:CURRent[:RESult]?
FETCh:POWer:P2DB:OUT:MAXimum[:RESult]?
FETCh:POWer:P2DB:OUT:MINimum[:RESult]?
FETCh:POWer:P2DB:OUT:CURRent[:RESult]?
FETCh:POWer:P3DB:OUT:MAXimum[:RESult]?
FETCh:POWer:P3DB:OUT:MINimum[:RESult]?
FETCh:POWer:P3DB:OUT: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 "Output Compression Point (1 dB / 2 dB / 3 dB)" on page 42

FETCh:POWer:SENSor:IN:MAXimum[:RESult]?
FETCh:POWer:SENSor:IN:MINimum[:RESult]?
FETCh:POWer:SENSor:IN:CURRent[:RESult]?

Fetches the signal power at the input power sensor.

Return values:

<Power> <numeric value>

Example: `FETC:POW:SENS:IN:CURR:RES?`

Usage: Query only

FETCh:POWer:SENSor:OUT:MAXimum[:RESult]?
FETCh:POWer:SENSor:OUT:MINimum[:RESult]?
FETCh:POWer:SENSor:OUT:CURRent[:RESult]?

Fetches the signal power at the output power sensor.

Return values:

<Power> <numeric value>

Example: `FETC:POW:SENS:OUT:CURR:RES?`

Usage: Query only

Manual operation: See "Power Out (Sensor)" on page 42

Retrieving baseband characteristics

FETCh:APAE:MAXimum[:RESult]?
FETCh:APAE:MINimum[:RESult]?
FETCh:APAE:CURRent[:RESult]?

This command queries the Average PAE (Power Added Efficiency) as shown in the Result Summary.

Return values:

<PAE> <numeric value>

Minimum, maximum or current Average PAE, depending on the command syntax.

Default unit: %

Example:
```
FETC:APAE:CURR?
```
would return, e.g.
```
1.231
```

Usage: Query only

Manual operation: See "Average PAE" on page 44

FETCh:BBPower:MAXimum[:RESult]?
FETCh:BBPower:MINimum[:RESult]?
FETCh:BBPower:CURRent[:RESult]?

This command queries the measured baseband power (I_cc * V_cc) as shown in the Result Summary.

Return values:

<Power> <numeric value>

Minimum, maximum or current power, depending on the command syntax.

Default unit: W

Example:
```
FETC:BBP:CURR?
```
would return, e.g.
```
0.75
```

Usage: Query only

Manual operation: See "Power" on page 44

FETCh:ICC:MAXimum[:RESult]?
FETCh:ICC:MINimum[:RESult]?
FETCh:ICC:CURRent[:RESult]?

This command queries the measured baseband current (I_cc) as shown in the Result Summary.

Return values:

<Current> Minimum, maximum or current I_cc, depending on the command syntax.

Default unit: A

Example: `FETC:ICC:MAX?`
would return, e.g.
`2.63`

Usage: Query only

Manual operation: See "Current" on page 44

FETCh:IVOLtage:PURE:MAXimum[:RESult]?
FETCh:IVOLtage:PURE:MINimum[:RESult]?
FETCh:IVOLtage:PURE:CURRent[:RESult]?

This command queries the voltage measured at the baseband input I as shown in the Result Summary.

The returned value is a "pure" voltage that does not contain any correction factors.

Return values:

<Voltage> <numeric value>

Minimum, maximum or current voltage, depending on the command syntax.

Default unit: V

Example: `FETC:IVOL:PURE:CURR?`
would return, e.g.
`1.4`

Usage: Query only

Manual operation: See "Baseband Input Voltage I" on page 43

FETCh:QVOLtage:PURE:MAXimum[:RESult]?
FETCh:QVOLtage:PURE:MINimum[:RESult]?
FETCh:QVOLtage:PURE:CURRent[:RESult]?

This command queries the measured at the baseband input Q as shown in the Result Summary.

The returned value is a "pure" voltage that does not contain any correction factors.

Return values:

<Voltage> <numeric value>

Minimum, maximum or current voltage, depending on the command syntax.

Default unit: V

Example: `FETC:IVOL:PURE:CURR?`
would return, e.g.
`1.42`

Usage: Query only

Manual operation: See "Baseband Input Voltage Q" on page 43

FETCh:VCC:MAXimum[:RESult]?
FETCh:VCC:MINimum[:RESult]?
FETCh:VCC:CURRent[:RESult]?

This command queries the measured baseband voltage (V_cc) as shown in the Result Summary.

Return values:

<Current> Minimum, maximum or current voltage, depending on the command syntax.

Default unit: V

Example: `FETC:VCC:CURR?`
would return, e.g.
`0.4`

Usage: Query only

Manual operation: See "Voltage" on page 43

5.8.2.3 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 selected parameter.

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?`
//Result:
`0.244445,1e+007,-30,0.246109,2e+007,-30, -21.9096,...`

Usage: Query only

FETCh:PTABle:ACP:ACHannel<ch>:BALanced:MAXimum:X[:RESult]?
FETCh:PTABle:ACP:BALanced:MAXimum:X[:RESult]?
FETCh:PTABle:ACP:MAXimum:X[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:LOWer:MAXimum:X[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:UPPer:MAXimum:X[:RESult]?
FETCh:PTABle:AMAM:CWIDth:MAXimum:X[:RESult]?
FETCh:PTABle:AMPM:CWIDth:MAXimum:X[:RESult]?
FETCh:PTABle:BBPower:MAXimum:X[:RESult]?
FETCh:PTABle:CFACtor:MAXimum:X[:RESult]?
FETCh:PTABle:EVM:MAXimum:X[:RESult]?

FETCh:PTABle:ICC:MAXimum:X[:RESult]?
FETCh:PTABle:GAIN:MAXimum:X[:RESult]?
FETCh:PTABle:PAE:MAXimum:X[:RESult]?
FETCh:PTABle:P1DB:MAXimum:X[:RESult]?
FETCh:PTABle:P2DB:MAXimum:X[:RESult]?
FETCh:PTABle:P3DB:MAXimum:X[:RESult]?
FETCh:PTABle:POUT:MAXimum:X[:RESult]?
FETCh:PTABle:RMS:MAXimum:X[:RESult]?
FETCh:PTABle:VCC:MAXimum:X[:RESult]?

These commands query the x-axis value at which the maximum result value for the parameter was determined, as shown in the "Parameter Sweep" table.

For details on the parameters, see Chapter 2.2, "Amplifier parameters", on page 33.

Return values:

<Results> <numeric value>

The value depends on the parameter selected for the x-axis (see CONFigure:PSWeep:X:SETTing on page 307).

Example: `FETC:PTAB:VCC:MAX:X:RES?`

Usage: Query only

FETCh:PTABle:ACP:ACHannel<ch>:BALanced:MAXimum:Y[:RESult]?
FETCh:PTABle:ACP:BALanced:MAXimum:Y[:RESult]?
FETCh:PTABle:ACP:MAXimum:Y[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:LOWer:MAXimum:Y[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:UPPer:MAXimum:Y[:RESult]?
FETCh:PTABle:AMAM:CWIDth:MAXimum:Y[:RESult]?
FETCh:PTABle:AMPM:CWIDth:MAXimum:Y[:RESult]?
FETCh:PTABle:BBPower:MAXimum:Y[:RESult]?
FETCh:PTABle:CFACtor:MAXimum:Y[:RESult]?
FETCh:PTABle:EVM:MAXimum:Y[:RESult]?
FETCh:PTABle:ICC:MAXimum:Y[:RESult]?
FETCh:PTABle:GAIN:MAXimum:Y[:RESult]?
FETCh:PTABle:PAE:MAXimum:Y[:RESult]?
FETCh:PTABle:P1DB:MAXimum:Y[:RESult]?
FETCh:PTABle:P2DB:MAXimum:Y[:RESult]?
FETCh:PTABle:P3DB:MAXimum:Y[:RESult]?
FETCh:PTABle:POUT:MAXimum:Y[:RESult]?
FETCh:PTABle:RMS:MAXimum:Y[:RESult]?
FETCh:PTABle:VCC:MAXimum:Y[:RESult]?

These commands query the y-axis value at which the maximum result value for the parameter was determined, as shown in the "Parameter Sweep" table.

For details on the parameters, see Chapter 2.2, "Amplifier parameters", on page 33.

Return values:

<Results> <numeric value>

The value depends on the parameter selected for the y-axis (see CONFigure:PSWeep:Y:SETTing on page 308).

Example: `FETC:PTAB:VCC:MAX:Y:RES?`

Usage: Query only

FETCh:PTABle:ACP:ACHannel<ch>:BALanced:MAXimum[:RESult]?
FETCh:PTABle:ACP:BALanced:MAXimum[:RESult]?
FETCh:PTABle:ACP:MAXimum[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:LOWer:MAXimum[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:UPPer:MAXimum[:RESult]?
FETCh:PTABle:AMAM:CWIDth:MAXimum[:RESult]?
FETCh:PTABle:AMPM:CWIDth:MAXimum[:RESult]?
FETCh:PTABle:BBPower:MAXimum[:RESult]?
FETCh:PTABle:CFACtor:MAXimum[:RESult]?
FETCh:PTABle:EVM:MAXimum[:RESult]?
FETCh:PTABle:GAIN:MAXimum[:RESult]?
FETCh:PTABle:ICC:MAXimum[:RESult]?
FETCh:PTABle:PAE:MAXimum[:RESult]?
FETCh:PTABle:P1DB:MAXimum[:RESult]?
FETCh:PTABle:P2DB:MAXimum[:RESult]?
FETCh:PTABle:P3DB:MAXimum[:RESult]?
FETCh:PTABle:POUT:MAXimum[:RESult]?
FETCh:PTABle:RMS:MAXimum[:RESult]?
FETCh:PTABle:VCC:MAXimum[:RESult]?

These commands query the maximum result values for the parameter as shown in the "Parameter Sweep" table.

For details on the parameters, see Chapter 2.2, "Amplifier parameters", on page 33.

Return values:
<Results> <numeric value>

Example:
```
FETC:PTAB:VCC:MIN:X?
//Result: 10000000
```

Usage: Query only

FETCh:PTABle:ACP:ACHannel<ch>:BALanced:MINimum:X[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:LOWer:MINimum:X[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:UPPer:MINimum:X[:RESult]?
FETCh:PTABle:ACP:BALanced:MINimum:X[:RESult]?
FETCh:PTABle:ACP:MINimum:X[:RESult]?
FETCh:PTABle:AMAM:CWIDth:MINimum:X[:RESult]?
FETCh:PTABle:AMPM:CWIDth:MINimum:X[:RESult]?
FETCh:PTABle:BBPower:MINimum:X[:RESult]?
FETCh:PTABle:CFACtor:MINimum:X[:RESult]?
FETCh:PTABle:EVM:MINimum:X[:RESult]?
FETCh:PTABle:GAIN:MINimum:X[:RESult]?
FETCh:PTABle:ICC:MINimum:X[:RESult]?
FETCh:PTABle:PAE:MINimum:X[:RESult]?
FETCh:PTABle:P1DB:MINimum:X[:RESult]?

FETCh:PTABle:P2DB:MINimum:X[:RESult]?
FETCh:PTABle:P3DB:MINimum:X[:RESult]?
FETCh:PTABle:POUT:MINimum:X[:RESult]?
FETCh:PTABle:RMS:MINimum:X[:RESult]?
FETCh:PTABle:VCC:MINimum:X[:RESult]?

These commands query the x-axis value at which the minimum result value for the parameter was determined, as shown in the "Parameter Sweep" table.

For details on the parameters, see Chapter 2.2, "Amplifier parameters", on page 33.

Return values:

<Results>	<numeric value> The value depends on the parameter selected for the x-axis (see `CONFigure:PSWeep:X:SETTing` on page 307).
Example:	`FETC:PTAB:VCC:MIN:X:RES?`
Usage:	Query only

FETCh:PTABle:ACP:ACHannel<ch>:BALanced:MINimum:Y[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:LOWer:MINimum:Y[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:UPPer:MINimum:Y[:RESult]?
FETCh:PTABle:ACP:BALanced:MINimum:Y[:RESult]?
FETCh:PTABle:ACP:MINimum:Y[:RESult]?
FETCh:PTABle:AMAM:CWIDth:MINimum:Y[:RESult]?
FETCh:PTABle:AMPM:CWIDth:MINimum:Y[:RESult]?
FETCh:PTABle:BBPower:MINimum:Y[:RESult]?
FETCh:PTABle:CFACtor:MINimum:Y[:RESult]?
FETCh:PTABle:EVM:MINimum:Y[:RESult]?
FETCh:PTABle:GAIN:MINimum:Y[:RESult]?
FETCh:PTABle:ICC:MINimum:Y[:RESult]?
FETCh:PTABle:PAE:MINimum:Y[:RESult]?
FETCh:PTABle:P1DB:MINimum:Y[:RESult]?
FETCh:PTABle:P2DB:MINimum:Y[:RESult]?
FETCh:PTABle:P3DB:MINimum:Y[:RESult]?
FETCh:PTABle:POUT:MINimum:Y[:RESult]?
FETCh:PTABle:RMS:MINimum:Y[:RESult]?
FETCh:PTABle:VCC:MINimum:Y[:RESult]?

These commands query the y-axis value at which the minimum result value for the parameter was determined, as shown in the "Parameter Sweep" table.

For details on the parameters, see Chapter 2.2, "Amplifier parameters", on page 33.

Return values:

<Results>	<numeric value> The value depends on the parameter selected for the y-axis (see `CONFigure:PSWeep:Y:SETTing` on page 308).
Example:	`FETC:PTAB:VCC:MIN:Y:RES?`
Usage:	Query only

FETCh:PTABle:ACP:ACHannel<ch>:BALanced:MINimum[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:LOWer:MINimum[:RESult]?
FETCh:PTABle:ACP:ACHannel<ch>:UPPer:MINimum[:RESult]?
FETCh:PTABle:ACP:BALanced:MINimum[:RESult]?
FETCh:PTABle:ACP:MINimum[:RESult]?
FETCh:PTABle:AMAM:CWIDth:MINimum[:RESult]?
FETCh:PTABle:AMPM:CWIDth:MINimum[:RESult]?
FETCh:PTABle:BBPower:MINimum[:RESult]?
FETCh:PTABle:CFACtor:MINimum[:RESult]?
FETCh:PTABle:EVM:MINimum[:RESult]?
FETCh:PTABle:GAIN:MINimum[:RESult]?
FETCh:PTABle:ICC:MINimum[:RESult]?
FETCh:PTABle:PAE:MINimum[:RESult]?
FETCh:PTABle:P1DB:MINimum[:RESult]?
FETCh:PTABle:P2DB:MINimum[:RESult]?
FETCh:PTABle:P3DB:MINimum[:RESult]?
FETCh:PTABle:POUT:MINimum[:RESult]?
FETCh:PTABle:RMS:MINimum[:RESult]?
FETCh:PTABle:VCC:MINimum[:RESult]?

These commands query the minimum result values for the parameter as shown in the "Parameter Sweep" table.

For details on the parameters, see Chapter 2.2, "Amplifier parameters", on page 33.

Return values:

<Results> <numeric value>

Example:
```
FETC:PTAB:VCC:MIN:X?
//Result: 10000000
```

Usage: Query only

5.8.2.4 Retrieving results of the statistics table

Retrieving the results in the statistics table requires different commands for every result type.

For each parameter, you can query either the current value (default) or the following statistical values calculated for the capture buffer or the entire measurement:

- `AVER`: average of the results
- `MIN`: minimum of the results
- `MAX`: maximum of the results
- `SDEV`: standard deviation of the results

For each result query, you can specify for which result range you require results:

- `SELected`: retrieve result of the currently selected result range
- `CURRent`: retrieve results over the current capture buffer
- `ALL`: retrieve the results over the entire measurement

In the following example, the SCPI commands querying the statistical results for amplitude droop are described.

- `FETCh:STABle:ADRoop:SELected[:RESult?]` queries the result of the currently selected result range (corresponds to the blue area of the statistics table).
- `FETCh:STABle:ADRoop:AVERage? CURRent` queries the average value of all result ranges found in the current capture buffer (corresponding to the green area of the statistics table).
- `FETCh:STABle:ADRoop:AVERage? ALL` queries the average value of all result ranges found in the entire measurement (corresponding to the black area of the statistics table).

FETCh:STABle:ADRoop:SELected[:RESult]?

Returns the amplitude droop for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:ADRoop:AVERage? <QueryRange>
FETCh:STABle:ADRoop:MAXimum? <Power>
FETCh:STABle:ADRoop:MINimum? <Power>
FETCh:STABle:ADRoop:STDeviation? <Power>

Returns the statistical value for the amplitude droop.

Query parameters:
<Power> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Amplitude Droop" on page 34

FETCh:STABle:AMAM:CWIDth:SELected[:RESult]?

Returns the AM/AM curve width for the currently selected result range.

Return values:
<CurveWidth>

Usage: Query only

FETCh:STABle:AMAM:CWIDth:AVERage? <QueryRange>
FETCh:STABle:AMAM:CWIDth:MAXimum? <QueryRange>
FETCh:STABle:AMAM:CWIDth:MINimum? <QueryRange>
FETCh:STABle:AMAM:CWIDth:STDeviation? <QueryRange>

Returns the statistical value for the AM/AM curve width.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "AM/AM Curve Width" on page 38

FETCh:STABle:AMPM:CWIDth:SELected[:RESult]?

Returns the AM/PM curve width for the currently selected result range.

Return values:
<CurveWidth>

Usage: Query only

FETCh:STABle:AMPM:CWIDth:AVERage? <QueryRange>
FETCh:STABle:AMPM:CWIDth:MAXimum? <QueryRange>
FETCh:STABle:AMPM:CWIDth:MINimum? <QueryRange>
FETCh:STABle:AMPM:CWIDth:STDeviation? <QueryRange>

Returns the statistical value for the AM/PM curve width.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "AM/PM Curve Width" on page 39

FETCh:STABle:APAE:AVG:SELected[:RESult]?

Returns the average average PAE for the currently selected result range.

Return values:
<PAE>

Usage: Query only

FETCh:STABle:APAE:AVG:AVERage? <QueryRange>
FETCh:STABle:APAE:AVG:MAXimum? <QueryRange>
FETCh:STABle:APAE:AVG:MINimum? <QueryRange>
FETCh:STABle:APAE:AVG:STDeviation? <QueryRange>

Returns the statistical value for the average average PAE.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Average PAE" on page 44

FETCh:STABle:APAE:MAX:SELected[:RESult]?

Returns the maximum average PAE for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:APAE:MAX:AVERage? <QueryRange>
FETCh:STABle:APAE:MAX:MAXimum? <QueryRange>
FETCh:STABle:APAE:MAX:MINimum? <QueryRange>
FETCh:STABle:APAE:MAX:STDeviation? <QueryRange>

Returns the statistical value for the maximum average PAE.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Average PAE" on page 44

FETCh:STABle:APAE:MIN:SELected[:RESult]?

Returns the minimum average PAE for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:APAE:MIN:AVERage? <QueryRange>
FETCh:STABle:APAE:MIN:MAXimum? <QueryRange>
FETCh:STABle:APAE:MIN:MINimum? <QueryRange>
FETCh:STABle:APAE:MIN:STDeviation? <QueryRange>

Returns the statistical value for the minimum average PAE.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Average PAE" on page 44

FETCh:STABle:BBPower:AVG:SELected[:RESult]?

Returns the average power for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:BBPower:AVG:AVERage? <QueryRange>
FETCh:STABle:BBPower:AVG:MAXimum? <QueryRange>
FETCh:STABle:BBPower:AVG:MINimum? <QueryRange>
FETCh:STABle:BBPower:AVG:STDeviation? <QueryRange>

Returns the statistical value for the average power.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Power" on page 44

FETCh:STABle:BBPower:MAX:SELected[:RESult]?

Returns the maximum power for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:BBPower:MAX:AVERage? <Power>
FETCh:STABle:BBPower:MAX:MAXimum? <QueryRange>

FETCh:STABle:BBPower:MAX:MINimum? <QueryRange>
FETCh:STABle:BBPower:MAX:STDeviation? <QueryRange>

Returns the statistical value for the maximum power.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Power" on page 44

FETCh:STABle:BBPower:MIN:SELected[:RESult]?

Returns the minimum power for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:BBPower:MIN:AVERage? <Power>
FETCh:STABle:BBPower:MIN:MAXimum? <Power>
FETCh:STABle:BBPower:MIN:MINimum? <Power>
FETCh:STABle:BBPower:MIN:STDeviation? <Power>

Returns the statistical value for the minimum power.

Query parameters:
<Power> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Power" on page 44

FETCh:STABle:CFACtor:IN:SELected[:RESult]?

Returns the crest factor in for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:CFACtor:IN:AVERage? <QueryRange>
FETCh:STABle:CFACtor:IN:MAXimum? <QueryRange>

FETCh:STABle:CFACtor:IN:MINimum? <QueryRange>
FETCh:STABle:CFACtor:IN:STDeviation? <QueryRange>

Returns the statistical value for the crest factor in.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Crest Factor In" on page 41

FETCh:STABle:CFACtor:OUT:SELected[:RESult]?

Returns the crest factor out for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:CFACtor:OUT:AVERage? <QueryRange>
FETCh:STABle:CFACtor:OUT:MAXimum? <QueryRange>
FETCh:STABle:CFACtor:OUT:MINimum? <QueryRange>
FETCh:STABle:CFACtor:OUT:STDeviation? <QueryRange>

Returns the statistical value for the crest factor out.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Crest Factor Out" on page 41

FETCh:STABle:FERRor:SELected[:RESult]?

Returns the frequency error for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:FERRor:AVERage? <Error>
FETCh:STABle:FERRor:MAXimum? <QueryRange>

FETCh:STABle:FERRor:MINimum? <QueryRange>
FETCh:STABle:FERRor:STDeviation? <QueryRange>

Returns the statistical value for the frequency error.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Frequency Error" on page 34

FETCh:STABle:GAIN:SELected[:RESult]?

Returns the gain for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:GAIN:AVERage? <QueryRange>
FETCh:STABle:GAIN:MAXimum? <QueryRange>
FETCh:STABle:GAIN:MINimum? <QueryRange>
FETCh:STABle:GAIN:STDeviation? <QueryRange>

Returns the statistical value for the gain.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Gain" on page 41

FETCh:STABle:GIMBalance:SELected[:RESult]?

Returns the gain imbalance for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:GIMBalance:AVERage? <QueryRange>
FETCh:STABle:GIMBalance:MAXimum? <QueryRange>

FETCh:STABle:GIMBalance:MINimum? <QueryRange>
FETCh:STABle:GIMBalance:STDeviation? <QueryRange>

Returns the statistical value for the gain imbalance.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Gain Imbalance" on page 34

FETCh:STABle:ICC:AVG:SELected[:RESult]?

Returns the average current for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:ICC:AVG:AVERage? <QueryRange>
FETCh:STABle:ICC:AVG:MAXimum? <QueryRange>
FETCh:STABle:ICC:AVG:MINimum? <QueryRange>
FETCh:STABle:ICC:AVG:STDeviation? <QueryRange>

Returns the statistical value for the average current.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Current" on page 44

FETCh:STABle:ICC:MAX:SELected[:RESult]?

Returns the maximum current for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:ICC:MAX:AVERage? <QueryRange>
FETCh:STABle:ICC:MAX:MAXimum? <QueryRange>

FETCh:STABle:ICC:MAX:MINimum? <QueryRange>
FETCh:STABle:ICC:MAX:STDeviation? <QueryRange>

Returns the statistical value for the maximum current.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Current" on page 44

FETCh:STABle:ICC:MIN:SELected[:RESult]?

Returns the minimum current for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:ICC:MIN:AVERage? <QueryRange>
FETCh:STABle:ICC:MIN:MAXimum? <QueryRange>
FETCh:STABle:ICC:MIN:MINimum? <QueryRange>
FETCh:STABle:ICC:MIN:STDeviation? <QueryRange>

Returns the statistical value for the minimum current.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Current" on page 44

FETCh:STABle:IQIMbalance:SELected[:RESult]?

Returns the I/Q imbalance for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:IQIMbalance:AVERage? <QueryRange>
FETCh:STABle:IQIMbalance:MAXimum? <QueryRange>

FETCh:STABle:IQIMbalance:MINimum? <QueryRange>
FETCh:STABle:IQIMbalance:STDeviation? <QueryRange>

Returns the statistical value for the I/Q imbalance.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "I/Q Imbalance" on page 34

FETCh:STABle:IQOFfset:SELected[:RESult]?

Returns the I/Q offset for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:IQOFfset:AVERage? <QueryRange>
FETCh:STABle:IQOFfset:MAXimum? <QueryRange>
FETCh:STABle:IQOFfset:MINimum? <QueryRange>
FETCh:STABle:IQOFfset:STDeviation? <QueryRange>

Returns the statistical value for the I/Q offset.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "I/Q Offset" on page 35

FETCh:STABle:IVOLtage:AVG:SELected[:RESult]?

Returns the average baseband I input voltage for the currently selected result range.

Return values:
<Voltage>

Usage: Query only

FETCh:STABle:IVOLtage:AVG:AVERage? <Voltage>
FETCh:STABle:IVOLtage:AVG:MAXimum? <Voltage>

FETCh:STABle:IVOLtage:AVG:MINimum? <Voltage>
FETCh:STABle:IVOLtage:AVG:STDeviation? <Voltage>

Returns the statistical value for the average baseband voltage i.

Query parameters:
<Voltage> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Baseband Input Voltage I" on page 43

FETCh:STABle:IVOLtage:MAX:SELected[:RESult]?

Returns the maximum baseband I input voltage for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:IVOLtage:MAX:AVERage? <Voltage>
FETCh:STABle:IVOLtage:MAX:MAXimum? <Voltage>
FETCh:STABle:IVOLtage:MAX:MINimum? <Voltage>
FETCh:STABle:IVOLtage:MAX:STDeviation? <Voltage>

Returns the statistical value for the maximum baseband voltage i.

Query parameters:
<Voltage> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Baseband Input Voltage I" on page 43

FETCh:STABle:IVOLtage:MIN:SELected[:RESult]?

Returns the minimum baseband I input voltage for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:IVOLtage:MIN:AVERage? <Voltage>
FETCh:STABle:IVOLtage:MIN:MAXimum? <Voltage>

FETCh:STABle:IVOLtage:MIN:MINimum? <Voltage>
FETCh:STABle:IVOLtage:MIN:STDeviation? <Voltage>

Returns the statistical value for the minimum baseband voltage i.

Query parameters:

<Voltage> CURRent | ALL

Return values:

<Result>

Usage: Query only

Manual operation: See "Baseband Input Voltage I" on page 43

FETCh:STABle:MERRor:SELected[:RESult]?

Returns the magnitude error for the currently selected result range.

Return values:

<Power>

Usage: Query only

FETCh:STABle:MERRor:AVERage? <QueryRange>
FETCh:STABle:MERRor:MAXimum? <QueryRange>
FETCh:STABle:MERRor:MINimum? <QueryRange>
FETCh:STABle:MERRor:STDeviation? <QueryRange>

Returns the statistical value for the magnitude error.

Query parameters:

<QueryRange> CURRent | ALL

Return values:

<Result>

Usage: Query only

Manual operation: See "Magnitude Error" on page 35

FETCh:STABle:P1DB:IN:SELected[:RESult]?

Returns the 1dB input compression point for the currently selected result range.

Return values:

<Power>

Usage: Query only

FETCh:STABle:P1DB:IN:AVERage? <QueryRange>
FETCh:STABle:P1DB:IN:MAXimum? <QueryRange>

FETCh:STABle:P1DB:IN:MINimum? <QueryRange>
FETCh:STABle:P1DB:IN:STDeviation? <QueryRange>

Returns the statistical value for the 1dB input compression point.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Compression Point (1 dB / 2 dB / 3 dB)" on page 40

FETCh:STABle:P1DB:OUT:SELected[:RESult]?

Returns the 1dB output compression point for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:P1DB:OUT:AVERage? <QueryRange>
FETCh:STABle:P1DB:OUT:MAXimum? <QueryRange>
FETCh:STABle:P1DB:OUT:MINimum? <QueryRange>
FETCh:STABle:P1DB:OUT:STDeviation? <QueryRange>

Returns the statistical value for the 1dB output compression point.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Output Compression Point (1 dB / 2 dB / 3 dB)" on page 42

FETCh:STABle:P2DB:IN:SELected[:RESult]?

Returns the 2dB input compression point for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:P2DB:IN:AVERage? <QueryRange>
FETCh:STABle:P2DB:IN:MAXimum? <QueryRange>

FETCh:STABle:P2DB:IN:MINimum? <QueryRange>
FETCh:STABle:P2DB:IN:STDeviation? <QueryRange>

Returns the statistical value for the 2dB input compression point.

Query parameters:

<QueryRange> CURRent | ALL

Return values:

<Result>

Usage: Query only

Manual operation: See "Compression Point (1 dB / 2 dB / 3 dB)" on page 40

FETCh:STABle:P2DB:OUT:SELected[:RESult]?

Returns the 2dB output compression point for the currently selected result range.

Return values:

<Power>

Usage: Query only

FETCh:STABle:P2DB:OUT:AVERage? <QueryRange>
FETCh:STABle:P2DB:OUT:MAXimum? <QueryRange>
FETCh:STABle:P2DB:OUT:MINimum? <QueryRange>
FETCh:STABle:P2DB:OUT:STDeviation? <QueryRange>

Returns the statistical value for the 2dB output compression point.

Query parameters:

<QueryRange> CURRent | ALL

Return values:

<Result>

Usage: Query only

Manual operation: See "Output Compression Point (1 dB / 2 dB / 3 dB)" on page 42

FETCh:STABle:P3DB:IN:SELected[:RESult]?

Returns the 3dB input compression point for the currently selected result range.

Return values:

<Power>

Usage: Query only

FETCh:STABle:P3DB:IN:AVERage? <QueryRange>
FETCh:STABle:P3DB:IN:MAXimum? <QueryRange>

FETCh:STABle:P3DB:IN:MINimum? <QueryRange>
FETCh:STABle:P3DB:IN:STDeviation? <QueryRange>

Returns the statistical value for the 3dB input compression point.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Compression Point (1 dB / 2 dB / 3 dB)" on page 40

FETCh:STABle:P3DB:OUT:SELected[:RESult]?

Returns the 3dB output compression point for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:P3DB:OUT:AVERage? <QueryRange>
FETCh:STABle:P3DB:OUT:MAXimum? <QueryRange>
FETCh:STABle:P3DB:OUT:MINimum? <QueryRange>
FETCh:STABle:P3DB:OUT:STDeviation? <QueryRange>

Returns the statistical value for the 3dB output compression point.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Output Compression Point (1 dB / 2 dB / 3 dB)" on page 42

FETCh:STABle:PC:SELected[:RESult]?

Returns the average power consumption for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:PC:AVERage? <QueryRange>
FETCh:STABle:PC:MAXimum? <QueryRange>

FETCh:STABle:PC:MINimum? <QueryRange>
FETCh:STABle:PC:STDeviation? <QueryRange>

Returns the statistical value for the power consumption.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

FETCh:STABle:PCPA:SELected[:RESult]?

Returns the PC based average PAE for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:PCPA:AVERage? <QueryRange>
FETCh:STABle:PCPA:MAXimum? <QueryRange>
FETCh:STABle:PCPA:MINimum? <QueryRange>
FETCh:STABle:PCPA:STDeviation? <QueryRange>

Returns the statistical value for the PC based average PAE.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

FETCh:STABle:PERRor:SELected[:RESult]?

Returns the phase error for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:PERRor:AVERage? <QueryRange>
FETCh:STABle:PERRor:MAXimum? <QueryRange>
FETCh:STABle:PERRor:MINimum? <QueryRange>
FETCh:STABle:PERRor:STDeviation? <QueryRange>

Returns the statistical value for the phase error.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Phase Error" on page 35

FETCh:STABle:POWer:INPut:AVG:SELected[:RESult]?

Returns the average power in for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:POWer:INPut:AVG:AVERage? <QueryRange>
FETCh:STABle:POWer:INPut:AVG:MAXimum? <QueryRange>
FETCh:STABle:POWer:INPut:AVG:MINimum? <QueryRange>
FETCh:STABle:POWer:INPut:AVG:STDeviation? <QueryRange>

Returns the statistical value for the average power in.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Power In" on page 42

FETCh:STABle:POWer:INPut:MAX:SELected[:RESult]?

Returns the maximum power in for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:POWer:INPut:MAX:AVERage? <QueryRange>
FETCh:STABle:POWer:INPut:MAX:MAXimum? <QueryRange>
FETCh:STABle:POWer:INPut:MAX:MINimum? <QueryRange>
FETCh:STABle:POWer:INPut:MAX:STDeviation? <QueryRange>

Returns the statistical value for the maximum power in.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Power In" on page 42

FETCh:STABle:POWer:INPut:MIN:SELected[:RESult]?

Returns the minimum power in for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:POWer:INPut:MIN:AVERage? <QueryRange>
FETCh:STABle:POWer:INPut:MIN:MAXimum? <QueryRange>
FETCh:STABle:POWer:INPut:MIN:MINimum? <QueryRange>
FETCh:STABle:POWer:INPut:MIN:STDeviation? <QueryRange>

Returns the statistical value for the minimum power in.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Power In" on page 42

FETCh:STABle:POWer:OUTPut:AVG:SELected[:RESult]?

Returns the average power out for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:POWer:OUTPut:AVG:AVERage? <QueryRange>
FETCh:STABle:POWer:OUTPut:AVG:MAXimum? <QueryRange>
FETCh:STABle:POWer:OUTPut:AVG:MINimum? <QueryRange>
FETCh:STABle:POWer:OUTPut:AVG:STDeviation? <QueryRange>

Returns the statistical value for the average power out.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Power Out" on page 42

FETCh:STABle:POWer:OUTPut:MAX:SELected[:RESult]?

Returns the maximum power out for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:POWer:OUTPut:MAX:AVERage? <QueryRange>
FETCh:STABle:POWer:OUTPut:MAX:MAXimum? <QueryRange>
FETCh:STABle:POWer:OUTPut:MAX:MINimum? <QueryRange>
FETCh:STABle:POWer:OUTPut:MAX:STDeviation? <QueryRange>

Returns the statistical value for the maximum power out.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Power Out" on page 42

FETCh:STABle:POWer:OUTPut:MIN:SELected[:RESult]?

Returns the minimum power out for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:POWer:OUTPut:MIN:AVERage? <QueryRange>
FETCh:STABle:POWer:OUTPut:MIN:MAXimum? <QueryRange>
FETCh:STABle:POWer:OUTPut:MIN:MINimum? <QueryRange>
FETCh:STABle:POWer:OUTPut:MIN:STDeviation? <QueryRange>

Returns the statistical value for the minimum power out.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Power Out" on page 42

FETCh:STABle:QERRor:SELected[:RESult]?

Returns the quadrature error for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:QERRor:AVERage? <QueryRange>
FETCh:STABle:QERRor:MAXimum? <QueryRange>
FETCh:STABle:QERRor:MINimum? <QueryRange>
FETCh:STABle:QERRor:STDeviation? <QueryRange>

Returns the statistical value for the quadrature error.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Quadrature Error" on page 36

FETCh:STABle:QVOLtage:AVG:SELected[:RESult]?

Returns the average baseband Q input voltage for the currently selected result range.

Return values:
<Voltage>

Usage: Query only

FETCh:STABle:QVOLtage:AVG:AVERage? <Voltage>
FETCh:STABle:QVOLtage:AVG:MAXimum? <Voltage>
FETCh:STABle:QVOLtage:AVG:MINimum? <Voltage>
FETCh:STABle:QVOLtage:AVG:STDeviation? <Voltage>

Returns the statistical value for the average baseband input voltage q.

Query parameters:
<Voltage> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Baseband Input Voltage Q" on page 43

FETCh:STABle:QVOLtage:MAX:SELected[:RESult]?

Returns the maximum baseband Q input voltage for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:QVOLtage:MAX:AVERage? <Voltage>
FETCh:STABle:QVOLtage:MAX:MAXimum? <Voltage>
FETCh:STABle:QVOLtage:MAX:MINimum? <Voltage>
FETCh:STABle:QVOLtage:MAX:STDeviation? <Voltage>

Returns the statistical value for the maximum baseband input voltage q.

Query parameters:
<Voltage> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Baseband Input Voltage Q" on page 43

FETCh:STABle:QVOLtage:MIN:SELected[:RESult]?

Returns the minimum baseband Q input voltage for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:QVOLtage:MIN:AVERage? <Voltage>
FETCh:STABle:QVOLtage:MIN:MAXimum? <Voltage>
FETCh:STABle:QVOLtage:MIN:MINimum? <Voltage>
FETCh:STABle:QVOLtage:MIN:STDeviation? <Voltage>

Returns the statistical value for the minimum baseband input voltage q.

Query parameters:
<Voltage> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Baseband Input Voltage Q" on page 43

FETCh:STABle:REVM:AVG:SELected[:RESult]?

Returns the average raw EVM for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:REVM:AVG:AVERage? <QueryRange>
FETCh:STABle:REVM:AVG:MAXimum? <QueryRange>
FETCh:STABle:REVM:AVG:MINimum? <QueryRange>
FETCh:STABle:REVM:AVG:STDeviation? <QueryRange>

Returns the statistical value for the average raw evm.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Raw EVM" on page 37

FETCh:STABle:REVM:MAX:SELected[:RESult]?

Returns the maximum raw EVM for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:REVM:MAX:AVERage? <QueryRange>
FETCh:STABle:REVM:MAX:MAXimum? <QueryRange>
FETCh:STABle:REVM:MAX:MINimum? <QueryRange>
FETCh:STABle:REVM:MAX:STDeviation? <QueryRange>

Returns the statistical value for the maximum raw evm.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Raw EVM" on page 37

FETCh:STABle:REVM:MIN:SELected[:RESult]?

Returns the minimum raw EVM for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:REVM:MIN:AVERage? <QueryRange>
FETCh:STABle:REVM:MIN:MAXimum? <QueryRange>
FETCh:STABle:REVM:MIN:MINimum? <QueryRange>
FETCh:STABle:REVM:MIN:STDeviation? <QueryRange>

Returns the statistical value for the minimum raw evm.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Raw EVM" on page 37

FETCh:STABle:RMEV:AVG:SELected[:RESult]?

Returns the average raw model EVM for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:RMEV:AVG:AVERage? <QueryRange>
FETCh:STABle:RMEV:AVG:MAXimum? <QueryRange>
FETCh:STABle:RMEV:AVERage:MINimum? <QueryRange>
FETCh:STABle:RMEV:AVG:MINimum? <QueryRange>
FETCh:STABle:RMEV:AVG:STDeviation? <QueryRange>

Returns the statistical value for the average raw model evm.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Raw Model EVM" on page 37

FETCh:STABle:RMEV:MAX:SELected[:RESult]?

Returns the maximum raw model EVM for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:RMEV:MAX:AVERage? <QueryRange>
FETCh:STABle:RMEV:MAX:MAXimum? <QueryRange>
FETCh:STABle:RMEV:MAX:MINimum? <QueryRange>
FETCh:STABle:RMEV:MAX:STDeviation? <QueryRange>

Returns the statistical value for the maximum raw model evm.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Raw Model EVM" on page 37

FETCh:STABle:RMEV:MIN:SELected[:RESult]?

Returns the minimum raw model EVM for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:RMEV:MIN:AVERage? <QueryRange>
FETCh:STABle:RMEV:MIN:MAXimum? <QueryRange>
FETCh:STABle:RMEV:MIN:MINimum? <QueryRange>
FETCh:STABle:RMEV:MIN:STDeviation? <QueryRange>

Returns the statistical value for the minimum raw model evm.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Raw Model EVM" on page 37

FETCh:STABle:SRERor:SELected[:RESult]?

Returns the sample rate error for the currently selected result range.

Return values:
<Power>

Usage: Query only

FETCh:STABle:SRERor:AVERage? <QueryRange>
FETCh:STABle:SRERor:MAXimum? <QueryRange>
FETCh:STABle:SRERor:MINimum? <QueryRange>
FETCh:STABle:SRERor:STDeviation? <QueryRange>

Returns the statistical value for the sample rate error.

Query parameters:
<QueryRange> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Sample Rate Error" on page 37

FETCh:STABle:VCC:AVG:SELected[:RESult]?

Returns the average amplifier supply voltage for the currently selected result range.

Return values:
<Voltage>

Usage: Query only

FETCh:STABle:VCC:AVG:AVERage? <Voltage>
FETCh:STABle:VCC:AVG:MAXimum? <Voltage>
FETCh:STABle:VCC:AVG:MINimum? <Voltage>
FETCh:STABle:VCC:AVG:STDeviation? <Voltage>

Returns the statistical value for the average amplifier supply voltage.

Query parameters:
<Voltage> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Voltage" on page 43

FETCh:STABle:VCC:MAX:SELected[:RESult]?

Returns the maximum amplifier supply voltage for the currently selected result range.

Return values:
<Voltage>

Usage: Query only

FETCh:STABle:VCC:MAX:AVERage? <Voltage>
FETCh:STABle:VCC:MAX:MAXimum? <Voltage>
FETCh:STABle:VCC:MAX:MINimum? <Voltage>
FETCh:STABle:VCC:MAX:STDeviation? <Voltage>

Returns the statistical value for the maximum amplifier supply voltage.

Query parameters:
<Voltage> CURRent | ALL

Return values:
<Result>

Example: `FETCh:STABle:VCC:MAX:STD? ALL`

Usage: Query only

Manual operation: See "Voltage" on page 43

FETCh:STABle:VCC:MIN:SELected[:RESult]?

Returns the minimum amplifier supply voltage for the currently selected result range.

Return values:
<Voltage>

Usage: Query only

FETCh:STABle:VCC:MIN:AVERage? <Voltage>
FETCh:STABle:VCC:MIN:MAXimum? <Voltage>
FETCh:STABle:VCC:MIN:MINimum? <Voltage>
FETCh:STABle:VCC:MIN:STDeviation? <Voltage>

Returns the statistical value for the minimum amplifier supply voltage.

Query parameters:
<Voltage> CURRent | ALL

Return values:
<Result>

Usage: Query only

Manual operation: See "Voltage" on page 43

5.8.3 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

TRACe:IQ:REF[:DATA]?

This command queries the reference trace I/Q data.

Example:
```
//Query reference trace I/Q data
TRAC:IQ:REF?
```

Usage: Query only

Manual operation: See "Reference signal information" on page 49

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>

BB
Queries the data captured on the optional analog baseband input.

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

5.9 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 FSW 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

5.10 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 FSW-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	
FETCh:POWer:CURRent[:RESult]	FETCh:BBPower:CURRent[:RESult]	
FETCh:POWer:MAXimum[:RESult]	FETCh:BBPower:MAXimum[:RESult]	
FETCh:POWer:MINimum[:RESult]	FETCh:BBPower:MINimum[:RESult]	
CONFigure:GENerator:IPConnection:LEDState	CONFigure:GENerator:CONNection:CSTate	

5.11 Programming examples FSW-K18M

The following programming examples for the FSW-K18M application show you how to apply a (generalized) memory polynomial model to an input vector and return the resulting output vector. The input vector and resulting output vector are both scaled to volts.

5.11.1 Programming example: applying coefficients manually

The following programming example for the FSW-K18M application shows you how to apply the coefficients manually.

```
function [vfcOutput,fRMSLevelOffsetdB]
 = ApplyCoefficients(vfcInput,vfcCoeffs,nKa,nLa,nKb,nLb,nMb,nKc,nLc,nMc)
 % ========================================================================
 % Copyright 2024 Rohde & Schwarz GmbH & Co. KG
 % Rohde & Schwarz products and services are supplied to customers subject
 % to certain contractual terms and conditions.
 % In addition, there are some requirements that apply especially to certain
 % products, customers or circumstances.
 % Detailed legal information for customers and users can be found here:
 %     http://www.termsofuse.rohde-schwarz.com
 % =========================================================================

 % This function applies a (generalized) memory polynomial model to an input vector and
 % returns the resulting output vector.

 % Input variables:
 %   vfcInput:       complex input vector in Volts
 %   vfcCoeffs:      complex coefficients,
 %                   the exported coefficients can be applied
 %                   without re-sorting,
 %                   the sorting of the elements can be found in the FSx-K18 manual
 %   nKa:            Maximum Polynomial Order of the aligned
 %                   terms
 %   nLa:            Maximum Memory Order of the aligned
 %                   terms
 %   nKb:            Maximum Polynomial Order of the lagging
 %                   cross-terms
 %   nLb:            Maximum Memory Order of the lagging
 %                   cross-terms
 %   nMb:            Maximum Cross-term Order of the lagging
 %                   cross-terms
 %   nKc:            Maximum Polynomial Order of the leading
 %                   cross-terms
 %   nLc:            Maximum Memory Order of the leading
 %                   cross-terms
 %   nMc:            Maximum Cross-term Order of the leading
```

```
%                   cross-terms



%check size of coeffs and configured parameters
nExpectedSize=nKa*(nLa+1)+(nKb-1)*(nLb+1)*nMb+(nKc-1)*(nLc+1)*nMc;
if length(vfcCoeffs)==nExpectedSize

  % Precompute poly orders
  nKMax=max([nKa,nKb,nKc]);
  vfcMagPoly = (abs(vfcInput).^(1:nKMax-1));

  % Poly, no memory
  vfcPolyVector = [vfcInput,vfcInput.*vfcMagPoly(:,1:nKa-1)];

  vfcXVector=[];
  if nLa > 0
    % lag memory, add left
    for n=nLa:-1:1
       vfcXVector = [circshift(vfcPolyVector,n,1),vfcXVector];
    end
    %add the part without only NL and no memory at the beginning
    vfcXVector =[vfcPolyVector,vfcXVector];
  else %%if only poly order and no memory is selected
    vfcXVector=vfcPolyVector;
  end


  if nMb > 0
    % lagging cross terms
    for n=0:nLb
        for m=1:nMb
            vfcXVector = [vfcXVector,circshift(vfcInput,n).
            *circshift(vfcMagPoly(:,1:nKb-1),n+m)];

        end
    end
  end

  if nMc > 0
    % leading cross terms
    for n=0:nLc
       for m=1:nMc
            vfcXVector = [vfcXVector,circshift(vfcInput,n).
            *circshift(vfcMagPoly(:,1:nKc-1),n-m)];
        end
    end
  end

  % Apply coefficients, vfcOutput is the predistorted signal
```

```
  vfcOutput = vfcXVector *  vfcCoeffs;

  %Compute the change of the RMS level for the new waveform in comparison to the input
  fInputNorm=norm(vfcInput);
  fOutputNorm=norm(vfcOutput);

  fRMSLevelOffsetdB = 20*log10(fOutputNorm / fInputNorm);
  disp(['RMS Level Change required at the generator:
     ',num2str(fRMSLevelOffsetdB),' dBm.'])

 else
  disp('Number of Coefficients does not match the configuration')
 end
end
```

5.11.2 Programming example: applying coefficients from file

The following programming example for the FSW-K18M application shows you how to apply the coefficients from a file.

The programming example reads the coefficients, as well as the orders for polynomial order, memory order and cross terms from the specified `csv` file. The formatting of the file needs to be the same as the coefficients `csv` files that can be saved in K18M.

```
function [vfcOutput,fRMSLevelOffsetdB]
= ApplyCoefficientsFromFile(vfcInput,sCoefficientsFileName)
 % =========================================================================
 % Copyright 2024 Rohde & Schwarz GmbH & Co. KG
 % Rohde & Schwarz products and services are supplied to customers subject
 % to certain contractual terms and conditions.
 % In addition, there are some requirements that apply especially to certain
 % products, customers or circumstances.
 % Detailed legal information for customers and users can be found here:
 %     http://www.termsofuse.rohde-schwarz.com
 % ==========================================================================

 % This function applies a (generalized) memory polynomial model to an input vector and
 % returns the resulting output vector.

 % Input variables:
 %   vfcInput:                  complex input vector in Volts
 %   sCoefficientsFileName:     Path of the file containing the coefficients,
 %                              as well as the orders for Polynomial
 %                              Order, Memory Order and Cross Terms


 [vfcCoeffs,nKa,nLa,nKb,nLb,nMb,nKc,nLc,nMc]=
 ReadCoeffsFromCSV(sCoefficientsFileName);
```

```
 %apply the coefficients from file
 [vfcOutput,fRMSLevelOffsetdB]
 = ApplyCoefficients(vfcInput,vfcCoeffs,nKa,nLa,nKb,nLb,nMb,nKc,nLc,nMc);

end


function [vfcCoeffs,nKa,nLa,nKb,nLb,nMb,nKc,nLc,nMc]
=ReadCoeffsFromCSV(sCoefficientsFileName)
 % this function reads the coefficients, as well as the orders for Polynomial Order,
 % Memory Order and Cross Terms from the specified csv file
 % the formatting of the file needs to be the same as the coefficients
 % csv failes that can be saved in K18M

 inputTable=readtable(sCoefficientsFileName);

 %read out the coefficients
 vfcCoeffs=inputTable{:,5}+1j*inputTable{:,6};

 %read out the vectors to specify the orders for the coefficients
 viK=inputTable{:,1};
 viL=inputTable{:,2};
 viMb=inputTable{:,3};
 viMc=inputTable{:,4};

 %determine the used nKa,nLa,nKb,nLb,nMb,nKc,nLc,nMc
 nMb=max(viMb);
 nMc=max(viMc);

 IndEndAlignedPart=find(viMb~=0,1)-1;
 nKa=viK(IndEndAlignedPart);
 nLa=viL(IndEndAlignedPart);

 IndEndLagPart=find(viMc~=0,1)-1;
 nKb=viK(IndEndLagPart);
 nLb=viL(IndEndLagPart);

 IndEndLeadPart=size(inputTable,1);
 nKc=viK(IndEndLeadPart);
 nLc=viL(IndEndLeadPart);

end
```

List of Commands (Amplifier)

Index

A

B

C

D

U

V

W

X

Y

Z