Programming Manual

RFM3000 Series
RF Power Meter

bkprecision.com

BK PRECISION

Contents

About Commands & Queries

This section lists and describes the remote control commands and queries recognized by the instrument. All commands and queries can be executed in either local or remote state.

The description, command syntax, query syntax, example and respond can be found in a section. The commands are given in both long and short form. All examples are shown in short form. Queries perform actions such as obtaining information, and are recognized by the question mark (?) following the header.

1.1 How They are Listed

The commands are listed by subsystem and alphabetical order according to their short form.

1.2 How They are Described

In the descriptions themselves, a brief explanation of the function performed is given. This is followed by a presentation of the formal syntax, with the header given in Upper-and-Lower-Case characters and the short form derived from it in ALL UPPER-CASE characters. Where applicable, the syntax of the query is given with the format of its response.

1.3 When can they be used?

The commands and queries listed here can be used for RFM3000 series.

1.4 Command Notation

The following notation is used in the commands:

< > Angular brackets enclose words that are used as placeholders, of which there are two types: the header path and the data parameter of a command.

:= A colon followed by an equals sign separates a placeholder from the description of the type and range of values that may be used in a command instead of the placeholder.

{ } Braces enclose a list of choices, one of which one must be made.

[ ] Square brackets enclose optional items.

... An ellipsis indicates that the items both to its left and right may be repeated a number of times.

Common Command Introduction

IEEE standard defines the common commands used for querying the basic inSyntaxion of the instrument or executing basic operations. These commands usually start with "*" and the length of the keywords of the command is usually 3 characters.

Short	Subsystem	What Command/Query does
*IDN	SYSTEM	Returns a string that uniquely identifies the instrument.
*OPC	SYSTEM	Generates the OPC message in the standard event status register when all pending overlapped operations have been completed.
*OPC?	SYSTEM	Returns an ASCII "+1" when all pending overlapped operations have been completed.
*TST?	SYSTEM	Returns the result of the self-test.
*WAI	SYSTEM	Prohibits the instrument from executing any new commands until all pending overlapped commands have been completed

2.1 *IDN?

Description The *IDN? query causes the instrument to identify itself. The response comprises manufacturer, model, serial number, software version and firmware version.

Query Syntax *IDN?

Response Syntax *IDN, <Manufacturer>,<model>,<serial number>, <firmware version>, <hardware version>.

< Manufacturer>:="BK" is used to identify the manufacturer.

<model>:= A model identifier less than 14 characters will contain the model number.

<serial number>:= Each product has its own number, the serial number labels the product uniqueness.

<firmware version>:= A serial numbers about software version.

<hardware version>:=The hardware level field, should contain inSyntaxion about all separately revisable subsystems. This inSyntaxion can be contained in single or multiple revision codes.

Example *IDN?

Returns: B&K Precision,4088,XXXXXXXX,1.2.5

2.2 *OPC

Description The *OPC (Operation Complete) command sets the OPC bit (bit 0) in the standard Event Status Register (ESR). This command has no other effect on the operation of the device because the instrument starts parsing a command or query only after it has completely processed the previous command or query.
The query is not a true query - a value of zero will never be returned. The *OPC? query will always responds with the ASCII character 1 because the device only responds to the query when the previous command has been entirely executed.

Command Syntax *OPC

Query Syntax *OPC?

Response Syntax *OPC 1

2.3 *TST?

Description The *TST? query performs an internal self-test and the response indicates whether the self-test has detected any errors. The self-test includes testing the hardware of all channels. Hardware failures are identified by a unique binary code in the returned <status> number. A "0" response indicates that no failures occurred or a sined,16-bit number if any errors are detected.

Query Syntax *TST?

Query Response *TST <status>
<status> := 0 self-test successful

Example TST?

Return(if no failure): 0

2.4 *WAI

Description The *WAI (WAIT to continue) command, requires by the IEEE 488.2 standard, has no effect on the instrument, as the signal generator only starts processing a command when the previous command has been entirely executed. The RF Power Meter always operates in non-overlapped, sequential mode, therefore this command is accepted as valid, but takes no action.

Command Syntax *WAI

Related Commands *OPC

Calculate Subsystem

Functions in the calculate subsystem are used to configure the measurement mode and control which portions of the acquired measurement data is used and how it is processed to yield a finished measurement. In addition to measurement mode, calculate is used to define mathematical operations, and measurement units. The numeric suffix of the calculate program mnemonic in the CALCulate commands refers to a processing and display "channel", that is CALCulate1 and CALCulate2 represent the power meter's Channel 1 and Channel 2 functions. The CALCulate commands generally DO NOT affect the data acquisition portion of the measurement (see the **Sense Subsystem**). In a signal-flow block diagram, the CALCulate block operations will follow those of the SENSe block. Note that CALCulate commands will not generate an error if used with a disconnected or non-existing channel on the RF Power Meter.

3.1 CALCulate:DCYCle

Description Set or return the duty cycle correction factor currently in use on the selected channel.

Command Syntax CALCulate<channel>:CORRection:DCYCle <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := { 0.01 to 100.00 percent }

Query Format CALCulate<channel>:CORRection:DCYCle?

Example
```
CALC1:CORR:DCYC 50
CALC1:CORR:DCYC?
```

Query Respond Returns: 50

Note:

Only available in CW Sensor (Modulated Mode)

3.2 CALCulate:MODE

Description Set or return the system measurement mode. MODULATED mode is a continuous measurement mode primarily for continuously modulated or CW signals, PULSE mode is a signal triggered, oscilloscope-like mode that acquires and analyzes a pulsed signal as a series of one or more triggered sweeps, and STATISTICAL mode performs long-term power distribution analysis on modulated signals, and may be operated in a start-stop continuous mode, or a decimated continuous mode.
Note that the measurement mode is global and affects all channels. If any channel has a CW sensor installed, peak or statistical measurements will be unavailable, but the "primary " average power measurement will still be performed.

Command Syntax CALCulate:MODE <character data>
<character data> = MODulated, PULSe, STATistical
Valid Modes: Any

Query Format CALCulate:MODE?

Example
```
CALC:MODE MOD
CALC:MODE?
```

Query Respond Returns: MODulated

3.3 CALCulate:PKHLD

Description Set or return the operating mode of the selected channel's peak hold function. When set to OFF, instantaneous peaks are only held for a short time, and then decayed towards the average power at a rate proportional to the filter time. This is the best setting for most signals, because the peak will always represent the peak power of the current signal, and the resulting peak-to-average ratio will be correct shortly after any signal level changes.

When set to ON, instantaneous peaks are held until reset by a new INITiate command or cleared manually. This setting is used when it is desirable to hold the highest peak over a long measurement interval without any decay.

Command Syntax CALCulate<channel>:PKHLD <state>
<channel> := {1 |2 |3 |4}
<state> := { 0, 1, OFF, AVG, INST}

Query Format CALCulate<channel>:PKHLD?

Example CALC1:PKHLD AVG CALC1:PKHLD?

Query Respond Returns: AVG

Note:

Available in Modulated and Pulse Modes

3.4 CALCulate:STATe

Description Set or return the measurement state of the selected channel. When ON, the channel performs measurements; when OFF, the channel is disabled and no measurements are performed.

Command Syntax CALCulate<channel>:STATe <Boolean>
<channel> := {1|2|3|4}
<state> := {0 |1 or OFF |ON}

Query Format CALCulate<channel>:STATe?

Example CALC1:STAT ON
CALC1:STATe?

Query Respond Returns: 1

3.5 CALCulate:UNITs

Description Set or return units for the selected channel. For power sensors, voltage is calculated with reference to the sensor input impedance. Note that for ratiometric results, logarithmic units will always return dBr (dB relative) while linear units return percent.

Command Syntax CALCulate<channel>:UNITs <character data> <channel> := {1|2|3|4}
<character data> := {DBMw, Watts, Volts, DBV, DBMV, DBUV}

Query Format CALCulate<channel>:UNITs?

Example CALC1:UNIT Volts
CALC1:UNIT?

Query Respond Returns: Volts

Display Subsystem

The DISPlay group of commands is used to control the selection and presentation of textual, graphical and TRACe measurements.

4.1 DISPlay:CLEar

Description Clear the display traces and all the data buffers. Clears averaging filters too. Does NOT clear errors.

4.2 Command Syntax

DISPlay:CLEar

Example DISP:CLE

4.3 DISPlay:ENVELOPE[:STATe]

Description Enable/disable the Envelope mode.

Command Syntax DISPlay:ENVELOPE[:STATe] <Boolean>
<boolean> := { 0 |1 or OFF |ON }

Query Format DISPlay:ENVELOPE[:STATe]?

Example DISP:ENVE ON DISP:ENVE?

Query Respond Returns: 1

Note:

Available in **Pulse** and **Modulated** mode.

4.4 DISPlay:MODUlated:TIMEBASE

Description Set or return the Modulated Mode timebase in seconds/division. The PMX40 has fixed timebase settings in a 1-2-5 sequence. If the argument does not match one of these settings, it will be forced to the next highest entry. Optional units: minutes.

Command Syntax DISPlay:MODUlated:TIMEBASE <numeric_value>
<numeric_value> := {10e-9 to 10 s, (1-2-5 sequence), 30 s, 1, 2, 5, 10, 30, 60 min}

Query Format DISPlay:MODUlated:TIMEBASE?

Example DISP:MODU:TIMEBASE 1
DISP:MODU:TIMEBASE?

Query Respond Returns: 1 min

Note:

Available in **Modulated** mode. There are separate time bases for the modulated mode and the pulse mode.

4.5 DISPlay:MODUlated:TSPAN

Description Set or return the horizontal time span of the display in modulated mode. Time span = 10* Time/Division. Optional units: minutes.

Command Syntax DISPlay:MODUlated:TSPAN <numeric value>
<numeric value> := {10e-8 to 100 sec in a 1-2-5 sequence, 300 s, 10, 20, 50, 100, 300, 600 min}

Query Format DISPlay:MODUlated:TSPAN?

Example DISP:MODU:TSPAN 10
DISP:MODU:TSPAN?

Query Respond Returns: 10 mins

Note:

Available in **Modulated** mode. There are separate time bases for the modulated mode and the pulse mode.

4.6 DISPlay:PULSe:TIMEBASE

Description Set or return the Pulse Mode timebase in seconds/division. The power meter has fixed timebase settings in a 1-2-5 sequence. If the argument does not match one of these settings, it will be forced to the next highest entry. Optional units: minutes.

Command Syntax DISPlay:PULSe:TIMEBASE <numeric value>
<numeric value> := {10e-9 to 10 s, (1-2-5 sequence), 30 s, 1, 2, 5, 10, 30, 60 min}

Query Format DISPlay:PULSe:TIMEBASE?

Example DISP:PULS:TIMEBASE 60
DISP:PULS:TIMEBASE?

Query Respond Returns: 60

Note:

Available in **Pulse** mode. There are separate time bases for the modulated mode and the pulse mode.

4.7 DISPlay:PULSe:TSPAN

Description Set or return the horizontal time span of the display in pulse mode.
Optional units: minutes.

$$Time \quad span = 10 * \frac{Time}{Division}$$

Command Syntax DISPlay:PULSe:TSPAN <numeric value>
<numeric value> := {10e-8 to 100 sec in a 1-2-5 sequence, 300 s, 10, 20, 50, 100, 300, 600 min}

Query Format DISPlay:PULSe:TSPAN?

Example DISP:PULS:TSPAN 100
DISP:PULS:TSPAN?

Query Respond Returns: 100 min

Note:

Available in **Pulse** mode. There are separate time bases for the modulated mode and the pulse mode.

4.8 DISPlay:[TEXt:]LIN:RESolution

Description Set or return the display resolution for linear power and voltage readings. The number of significant digits displayed is equal to the argument. This command also sets the resolution of measurements returned in remote mode.

Command Syntax DISPlay:[TEXt:]LIN:RESolution <numeric value>
<numeric value> = 3 to 5

Query Format DISPlay:[TEXt:]LIN:RESolution?

Example DISP:LIN:RES 5
DISP:LIN:RES?

Query Respond Returns: 5

4.9 DISPlay:[TEXt:]LOG:RESolution

Description Set or return the display resolution for logarithmic power and voltage readings. The number of decimal places displayed is equal to the argument. This command also sets the resolution of measurements returned in remote mode.

Command Syntax DISPlay:[TEXt:]LOG:RESolution <numeric value>
<numeric value> := {0 to 3}

Query Format DISPlay:[TEXt:]LOG:RESolution?

Example DISP:LOG:RES 3
DISP:LOG:RES?

Query Respond Returns: 3

4.10 DISPlay:TRACe:HOFFSet

Description Set or return the statistical mode horizontal scale offset in dB. The offset value will appear at the leftmost edge of the scale with units dBr (decibels relative).

Command Syntax DISPlay:TRACe<channel>:HOFFSet <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := {-50.00 to +50.00 dB}

Query Format DISPlay:TRACe1:HOFFSet?

Example DISP:TRAC1:HOFFS 25 DISP:TRAC1:HOFFS?

Query Respond Returns: 25 dB

Note:

Available in **Statistical** mode.

4.11 DISPlay:TRACe:HSCALe

Description Set or return the statistical mode horizontal scale in dB/Div.

Command Syntax DISPlay:TRACe<channel>:HSCALe <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := {0.1 dB/Div to 5.0 dB/Div in a 1-2-5 sequence}

Query Format DISPlay:TRACe1:HSCALe?

Example DISP:TRAC1:HSCAL 1 DISP:TRAC1:HSCAL?

Query Respond Returns: 1 dB/Div

Note:

Available in **Statistical** mode.

4.12 DISPlay:TRACe:VCENTer

Description Set or return the power or voltage level of the horizontal centerline of the graph for the specified channel in channel units. If a change in the vertical scale causes the center maximum value to be exceeded, the center will be forced to the maximum value for the new range.

Command Syntax DISPlay:TRACe<channel>:VCENTer <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := {-200.00 to +200.00 dBm for dBm units}
+/- 10,000 times the vertical scale power/div for watts units
+/- 10,000 times the vertical scale volts/div for volts units

Query Format DISPlay:TRACe<channel>:VCENTer?

Example DISP:TRAC1:VCENT 10
DISP:TRAC1:VCENT?

Query Respond Returns: 10 dB

4.13 DISPlay:TRACe:VSCALe

Description Set or return the power or voltage vertical sensitivity of the trace display in channel units.

Command Syntax DISPlay:TRACe<channel>:VSCALe <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := {range in dB/division} |Units := dBm, dBV, dBmV, dBuV
<numeric value> := {range in watts/division} |Units := watts
<numeric value> := {range in volts/division} |Units := volts

Query Format DISPlay:TRACe<channel>:VSCALe?

Example DISP:TRAC1:VSCAL
DISP:TRAC1:VSCAL?

Query Respond Returns:

Fetch Subsystem

The FETCh? subsystem Returns specific measurement data from a measurement cycle that has been INITiated and is complete or free-running. FETCh? does not start a new measurement, so a series of FETCh? queries may be used to Returns more than one set of processed measurements from a complete set of acquired data. FETCh? usually Returns the current value of measurements, and should be used anytime free running data acquisition is taking place (INITiate:CONTinuous ON).

5.1 FETCh:ARRay:AMEAsure:POWer?

Description Returns an array of the current automatic amplitude measurements performed on a periodic pulse waveform. Measurements performed are: peak amplitude during the pulse, average amplitude over a full cycle of the pulse waveform, average amplitude during the pulse, IEEE top amplitude, IEEE bottom amplitude, overshoot, and droop.

The pulse "ON" interval used for peak and average calculations is defined by the SENSe:PULSe:STARTGT and :ENDGT time gating settings. A full pulse (rise and fall) must be visible on the display to make average and peak pulse power measurements, and a full cycle of the waveform must be visible to calculate average cycle amplitude.

Units are the same as the channel's units. The pulse overshoot is Returnsed in dB for logarithmic channel units, and percent for all other units.

Query Syntax FETCh<channel>:ARRay:AMEAsure:POWer?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC1, PulsePeak, CC2, PulseCycleAvg, CC3, PulseOnAvg, CC4, IEEE Top, CC5, IEEE Bot, CC6, Overshoot, CC7, Droop. Where the CCn's are the measurement condition codes for each measurement.

Note:

Available in **Pulse** mode only.
Timebase must be set appropriately to allow measurements (see above)

5.2 FETCh:ARRay:AMEAsure:STATistical?

Description Returns an array of the current automatic statistical measurements performed on a sample population. Measurements performed are: long term average, peak and minimum amplitude, peak-to-average ratio, amplitude at the CCDF percent cursor, statistical percent at the CCDF power cursor, and the sample population size in samples. The peak-to-average ratio is Returnsed in dB for logarithmic channel units, and percent for all other channel units.

Querry Syntax FETCh<channel>:ARRay:AMEAsure:STATistical?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC1, Pavg, CC2, Ppeak, CC3, Pmin, CC4, PkToAvgRatio, CC5, CursorPwr, CC6, CursorPct, CC7, Sample-Count Where the CCn's are the measurement condition codes for each measurement.

Note:

Available in **Statistical** mode only.

5.3 FETCh:ARRay:AMEAsure:TIMe?

Description Returns an array of the current automatic timing measurements performed on a periodic pulse waveform. Measurements performed are: the frequency, period, width, offtime and duty cycle of the pulse waveform, and the risetime and falltime of the edge transitions. For each of the measurements to be performed, the appropriate items to be measured must be visible on the display in GRAPH mode.

Pulse frequency, period, offtime and duty cycle measurements require that an entire cycle of the pulse waveform (minimum of three edge transitions) be present. Pulse width measurement requires that at least one full pulse is visible and is most accurate if the pulse width is at least 0.4 divisions (20 pixels). Risetime and falltime measurements require that the edge being measured is visible and will be most accurate if the transition takes at least 0.1 divisions (5 pixels).

It is always best to have the power meter set on the fastest timebase possible that meets the edge visibility restrictions. Set the trace averaging as high as practical to reduce fluctuations and noise in the pulse timing measurements. Note that the timing of the edge transitions is defined by the settings of the SENSe:PULSe:DISTal, :MESIal and :PROXimal settings; see the descriptions for those commands.

Units are the same as the channel's units.

Query Syntax FETCh<channel>:ARRay:AMEAsure:TIMe?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC1, PulseFreq, CC2, PulsePeriod, CC3, PulseWidth, CC4, Offtime, CC5, DutyCycle, CC6, Risetime, CC7, Falltime, CC8, EdgeDly, CC9, Skew in seconds Where the CCn's are the measurement condition codes for each measurement.

Note:

Available in **Pulse** mode only.
Timebase must be set appropriately to allow measurements (see above)

5.4 FETCh:ARRay:CW:POWer?

Description Returns the current average, maximum, minimum powers or voltages and the peak-toaverage ratio of the specified channel. Units are the same as the channel's units. The peak-to-average ratio is Returnsed in dB for logarithmic channel units, and percent for all other channel units.

The maximum and minimum power values will depend on the peak hold mode; see the description of the CALCulate:PKHLD command for details.

Query Syntax FETCh<channel>:ARRay:CW:POWer?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC1, Pavg, CC2, Pmax, CC3, Pmin, CC4, PkToAvgRatio Where the CCn's are the measurement condition codes for each measurement.

Note:

Available in **Pulse** and **Modulated** mode only.

5.5 FETCh:ARRay:MARKer:POWer?

Description Returns an array of the current marker measurements for the specified channel. The array consists of the average, maximum, and minimum power and peak-to-average ratio between the two markers, powers at both markers, and the ratio of the two markers.

The peak-to-average ratio and marker ratio are Returnsed in dB for log units, and percent for linear units.

Query Syntax FETCh<channel>:ARRay:MARKer:POWer?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC1, Pavg, CC2, Pmax, CC3, Pmin, CC4, PkToAvgRatio, CC5, Pwr@Marker1, CC6, Pwr@Marker2, CC7, Mrk1/Mrk2 ratio Where the CCn's are the measurement condition codes for each measurement.

Note:

Available in **Pulse** and **Modulated** mode only.

5.6 FETCh:CW:POWer?

Description Returns current average amplitude reading in channel units.

Querry Syntax FETCh<channel>:CW:POWer?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC, average power (watts, dBm) or average voltage (volts, dBv) Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode only.

5.7 FETCh:INTERval:AVERage?

Description Returns the average power or voltage in the time interval between marker 1 and marker 2 of the selected channel.

The units will be the same as the specified channel.

Query Syntax FETCh<channel>]:INTERval:AVERage?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC, average power or voltage between markers Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode only.

5.8 FETCh:INTERval:MAXFilt?

Description Returns the maximum filtered power or voltage in the time interval between marker 1 and marker 2 of the selected channel.

The units will be the same as the specified channel.

Query Syntax FETCh<channel>:INTERval:MAXFilt?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC, maximum filtered power or voltage between the markers Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode only.

5.9 FETCh:INTERval:MINFilt?

Description Returns the minimum filtered power or voltage in the time interval between marker 1 and marker 2 of the specified channel.

The units will be the same as the specified channel.

Query Syntax FETCh<channel>:INTERval:MINFilt?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC, minimum filtered power or voltage between markers Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode only.

5.10 FETCh:INTERval:MAXimum?

Description Returns the maximum instantaneous power or voltage in the time interval between marker 1 and marker 2 of the specified channel.

The units will be the same as the specified channel.

Query Syntax FETCh<channel>:INTERval:MAXimum?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC, maximum instantaneous power or voltage between markers Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode only.

5.11 FETCh:INTERval:MINimum?

Description Returns the minimum instantaneous power or voltage in the time interval between marker 1 and marker 2 of the specified channel.

The units will be the same as the specified channel.

Query Syntax FETCh<channel>:INTERval:MINimum?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC, minimum instantaneous power or voltage between markers Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode only.

5.12 FETCh:INTERval:PKAVG?

Description Returns the peak-to-average ratio of the power or voltage between marker 1 and marker 2 of the specified channel.

The units are dB for logarithmic channel units or percent for linear channel units.

Query Syntax FETCh<channel>:INTERval:PKAVG?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC, peak-to-average ratio of power or voltage between markers Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode only.

5.13 FETCh:MARKer:AVERage?

Description Returns the average power or voltage of the specified channel and marker.

The units are the same as the specified channel.

Query Syntax FETCh<channel>:MARKer<marker>:AVERage?
<channel> := {1 |2 |3 |4}
<marker> := {1 |2}

Query Response Returns: CC, average power or voltage at marker Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode only.

5.14 FETCh:MARKer:CURsor:PERcent?

Description Returns the CCDF cursor y-axis position in percent with respect to the value set by MARKer:POSItion:POWer (CCDF cursor mode is Power Ref). If CCDF cursor mode is Percent, Returns user setting. See MARKer:POSItion:POWer and MARKer:POSItion:PERcent. Refer to the RF Power Meter Instruction Manual Statistical Mode Automatic Measurements section for more information.

Query Syntax FETCh<channel>:MARKer:CURsor:PERcent?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC, percent CCDF Where CC is the measurement condition code.

Note:

Available in **Statistical** mode only.

5.15 FETCh:MARKer:CURsor:POWer?

Description Return the CCDF cursor x-axis position in relative power with respect to the value set by MARKer:POSItion:PERcent (CCDF cursor mode is Percent). If CCDF cursor mode is Power Ref, returns user setting. See MARKer:POSItion:POWer and MARKer:POSItion:PERcent. Refer to the RF Power Meter Instruction Manual Statistical Mode Automatic Measurements section for more information.

Query Syntax FETCh<channel>:MARKer:CURsor:POWer?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC, relative power (dBr) CCDF Where CC is the measurement condition code.

Note:

Available in **Statistical** mode only.

5.16 FETCh:MARKer:DELTa?

Description Returns the difference between MK1 and MK2 of the specified channel. The units will be the same as marker units.

Query Syntax FETCh<channel>:MARKer:DELTa?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC, (MK1 – MK2) Where CC is the measurement condition code.

5.17 FETCh:MARKer:MAXimum?

Description Returns the maximum power or voltage of the specified channel and marker. The units are the same as the specified channel.

Query Syntax FETCh<channel>:MARKer<marker>:MAXimum?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC, maximum power or voltage at marker Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode only.

5.18 FETCh:MARKer:MINimum?

Description Returns the minimum power or voltage of the specified channel and marker. The units will be the same as the specified channel.

Query Syntax FETCh<channel>:MARKer<marker>:MINimum?
<channel> := {1 |2 |3 |4}
<marker> := {1 |2}

Query Response Returns: CC, minimum power or voltage at marker Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode only.

5.19 FETCh:MARKer:RATio?

Description Returns the ratio of MK1 to MK2 of the specified channel. The units will be dB for logarithmic units or percent for linear units.

Query Syntax FETCh<channel>:Marker:RATio?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC, MK1/MK2 Where CC is the measurement condition code.

5.20 FETCh:MARKer:RDELTa?

Description Returns the difference between MK2 and MK1 of the specified channel. The units will be the same as marker units.

Query Syntax FETCh<channel>:MARKer:RDELTa?
S <channel> := {1 |2 |3 |4}

Query Response Returns: CC, MK2-MK1 Where CC is the measurement condition code.

FETCh:MA

5.21 RKer:RRATio?

Description Returns the ratio of MK2 to MK1 of the specified channel. The units will be dB for logarithmic units or percent for linear units.

Query Syntax FETCh<channel>:MARKer:RRATio?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC, MK2/MK1 Where CC is the measurement condition code.

5.22 FETCh:TEMPerature:AVERage?

Description Returns the average internal temperature of the specified power sensor.

Syntax FETCh<channel>:TEMPerature:AVERage?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC, Sensor<channel> Average Temp in degrees C Where CC is the measurement condition code.

Note:

Only available when using **RTP Sensors**.

5.23 FETCh:TEMPerature:CURRent?

Description Returns the current internal temperature of the specified power sensor.

Query Syntax FETCh<channel>:TEMPerature:CURRent?
<channel> := {1 |2 |3 |4}

Query Response Returns: CC, Sensor<channel> Temp in degrees C Where CCn is the measurement condition code.

Note:

Only available when using **RTP Sensors**.

Initiate & Abort Subsystem

The INITiate commands do not actually start measurements.They set a hardware trigger, which is then used to gate the actual measurements cycle.

6.1 ABORt

Description Terminates any measurement in progress and resets the state of the trigger system. The ABORt command will leave the measurement in a stopped condition with all current measurements cleared, and forces INITiate:CONTinuous to OFF.

Command Syntax ABORt

Example ABOR

6.2 INITiate:CONTinuous

Description Set or return the data acquisition mode for single or free-run measurements.
If INITiate:CONTinuous is set to ON, the power meter immediately begins taking measurements (Modulated, CW and Statistical Modes), or arms its trigger and takes a measurement each time a trigger occurs (Pulse Mode).
If set to OFF, the measurement will begin (or be armed) as soon as the INITiate command is issued, and will stop once the measurement criteria (averaging, filtering or sample count) has been satisfied.
INITiate:IMMediate and READ commands are invalid when INITiate:CONTinuous is set to ON. This convention does not result in a SCPI error.

Command Syntax INITiate:CONTinuous <state>
<state> := { 0 |1 or OFF |ON}

Query Format INITiate:CONTinuous?

Example INIT:CONT On

Query Respond Returns: 1

6.3 INITiate[:IMMediate[:ALL]]

Description Starts a single measurement cycle when INITiate:CONTinuous is set to OFF. In Modulated Mode, the measurement will complete once the power has been integrated for the full FILTer time. In Pulse Mode, enough trace sweeps must be triggered to satisfy the AVERaging setting. In Statistical Mode, acquisition stops once the terminal condition(s) are met. In each case, no reading will be returned until the measurement is complete.

This command is not valid when INITiate:CONTinuous is ON. This convention does not result in a SCPI error.

Command Syntax INITiate[:IMMediate[:ALL]]

Example INIT

Note:

For the command to function INITiate:CONTinuous must be set to OFF.

Marker Subsystem

The MARKer subsystem is used to configure and locate measurement markers (cursors) at specific points on the processed measurement waveform. FETCH? or READ? queries may then be used to retrieve measurements at the two markers and in the interval between them.

Markers are used in Pulse Mode to perform measurements at or between two time offsets relative to the trigger. In Statistical Mode markers measure the power at a particular statistical percent, or the percent at a specified power level. In Pulse Mode, the markers can only be placed on the visible portion of the trace (as defined by the timebase and trigger delay settings). Statistical Mode markers may be placed at any power or percent value and will still return readings.

7.1 MARKer:POSItion:PERcent

Description Set or return the percent probability (y-axis-position) of the CCDF cursor. Changing this setting will force the CCDF cursor mode to Percent for display purposes.

READ:MARKer:CURsor:POWer? and FETCh:MARKer:CURsor:POWer? commands are referenced to this value.

Command Syntax MARKer:POSItion:PERcent <numeric value>
<numeric value> := {0.000 to 100.000 % }

Query Format MARKer:POSItion:PERcent?

Example MARK:POSI:PER 50

Query Respond Returns: 50 %

Note:

Only available in **Statistical** mode.

7.2 MARKer:POSItion:POWer

Description Set or return the cumulative relative power (x-axis-position) of the CCDF cursor in dBr. Changing this setting will force the CCDF cursor mode to Power Ref for display purposes.

READ:MARKer:CURsor:PERcent? and FETCh:MARKer:CURsor:PERcent? commands are referenced to this value.

Syntax MARKer:POSItion:POWer <numeric value>
<numeric value> := {-100.000 to 100.000 dBr}

Query Format MARKer:POSItion:POWer?

Example MARK:POSI:POW 2
MARK:POSI:POW?

Query Respond Returns: 2 dB

Note:

Only available in **Statistical** mode.

7.3 MARKer:POSItion:TIMe

Description Set or return the time (x-axis-position) of the selected marker relative to the trigger. Time markers must be positioned within the time limits of the trace window in the graph display. If a time outside of the display limits is entered, the marker will be placed at the first or last time position as appropriate.

Syntax MARKer<marker>:POSItion:TIMe <numeric value> <marker> := {1 |2}
<numeric value> := { display-time in seconds (see restrictions)}

Query Format MARKer<marker>:POSItion:TIMe ?

Example MARK1:POSI:TIM 3
MARK1:POSI:TIM?

Query Respond Returns: 3 s

Note:

Available in **Pulse** and Modulated mode.
For zero delay trigger position in the center of the display , the following relationship must be satisfied:
TrigDly - (5*time/div) < MarkerTime < TrigDly + (5*time/div) where the timebase setting is time/div

Measure Subsystem

The measure subsystem is used to acquire data using a set of high level instructions. They are structured to allow the user to trade off fine control of the measurement process for easy operability.

MEASure? provides a complete capability where the power meter is configured, a measurement taken, and results returned in one operation. The instrument is set to a basic, predefined measurement state with little user intervention necessary or possible. When more precise control of measurement is required MEASure? should not be used. A sequence of configuration commands, generally from the CALCulate and SENSe groups should be used to set up the instrument for the measurement, then READ? or FETCH? commands are used to return the desired measurement data in a specific format.

8.1 MEASure:POWer?

Description Return average power using a default instrument configuration in Modulated mode and dBm units. Instrument remains stopped in Modulated Mode after a measurement.

Query Syntax Measure<channel>:POWer?
<channel> := {1 |2 |3 |4 }

Example MEAS1:POW?

Query Respond Returns: CC, Average power in dBm Where CC is the measurement condition code.

Note:

Automatically set to **Modulated** mode before measurement

8.2 MEASure:VOLTage?

Description Returns average voltage using a default instrument configuration in Modulated mode and volts units. Instrument remains stopped in Modulated mode after a measurement.

Query Syntax MEASure<channel>:VOLTage?
<channel> := {1 |2 |3 |4 }

Example MEAS1:VOLT?

Query Respond Returns: CC, Average voltage in linear volts Where CC is the measurement condition code.

Note:

Automatically set to **Modulated** mode before measurement.

Memory Subsystem

The memory subsystem reviews frequency dependent offset (FDOF) tables for the sensors.

9.1 MEMory:SNSR:CFFAST?

Description Return the sensor high bandwidth (FAST) frequency cal-factor table.

Query Syntax MEMory:SNSR<channel>:CFFAST?

Example MEM:SNSR1:CFFAST?

Query Respond Returns:

9.2 MEMory:SNSR:CFSLOW?

Description Return the sensor low bandwidth (SLOW) frequency cal-factor table.

Query Syntax MEMory:SNSR<channel>:CFSLOW?

Example MEM:SNSR1:CFSLOW?

Query Respond Returns:

Read Subsystem

The READ? subsystem initiates a measurement cycle, acquire data, and return specific measurement data. READ? performs the initiation, data acquisition, postprocessing, and data output portions of the measurement. READ? is equivalent to ABORting any operation in progress, INITiating a new measurement, then FETChing the data when it is ready.

READ? does not return data unless acquisition is complete. Since READ? initiates a new measurement every time it is issued, it should not be used for free running data acquisition (INITiate:CONTinuous ON).

For CW and Modulated Modes, the measurement is generally considered complete when the integration filter (see SENSe:FILTer) is filled. In Pulse Mode, the measurement is considered complete when all the number of complete traces specified by the SENSe:AVERage command have been acquired and averaged together. In Statistical Mode, the measurement is considered complete when the number of samples specified by TRIGger:CDF:COUNt has been gathered.

10.1 READ:ARRay:AMEAsure:POWer?

Description Returns an array of the current automatic amplitude measurements performed on a periodic pulse waveform. Measurements performed are: peak amplitude during the pulse, average amplitude over a full cycle of the pulse waveform, average amplitude during the pulse, IEEE top amplitude, IEEE bottom amplitude, overshoot, and droop. Units are the same as the channel's units. Note the pulse overshoot is returned in dB for logarithmic channel units, and percent for all other units.

The pulse "ON" interval used for peak and average calculations is defined by the SENSe:PULSe:STARTGT and :ENDGT time gating settings. A full pulse (rise and fall) must be visible on the display to make average and peak pulse power measurements. A full cycle of the waveform must be visible to calculate average cycle amplitude.

Query Syntax READ<channel>:ARRay:AMEAsure:POWer?
<channel> := {1 |2 |3 |4}

Example READ:ARR:AMEA:POW?

Query Response Returns: CC1, PulsePeak, CC2, PulseCycleAvg, CC3, PulseOnAvg, CC4, IEEE Top, CC5, IEEE Bot, CC6, Overshoot, CC7, Droop Where the CCn's are the measurement condition codes for each measurement.

Note:

Only available in **Pulse** mode.
Timebase must be set appropriately to allow measurements

10.2 READ:ARRay:AMEAsure:STATistical?

Description Returns an array of the current automatic statistical measurements performed on a sample population. Measurements performed are: long term average, peak and minimum amplitude, peak-to-average ratio, amplitude at the CCDF cursor, statistical percent at the CCDF cursor, and the sample population size in samples. The peakto-average ratio is returned in dB for logarithmic channel units, and percent for all other channel units.

Query Syntax READ:ARRay:AMEAsure:STATistical?
<channel> := {1 |2 |3 |4}

Example READ1:ARR:AMEA:STAT?

Query Response Returns: CC1, Pavg, CC2, Ppeak, CC3, Pmin, CC4, PkToAvgRatio, CC5, CursorPwr, CC6, CursorPct, CC7, Sample-Count Where the CCn's are the measurement condition codes for each measurement.

Note:

Only available in **Statistical** mode.

10.3 READ:ARRay:AMEAsure:TIMe?

Description Returns an array of the current automatic timing measurements performed on a periodic pulse waveform. Measurements performed are: the frequency, period, width, offtime and duty cycle of the pulse waveform, and the risetime and falltime of the edge transitions.

For each of the measurements to be performed, the appropriate items to be measured must be visible on the display in GRAPH mode. Pulse frequency, period, offtime and duty cycle measurements require that an entire cycle of the pulse waveform (minimum of three edge transitions) be present. Pulse width measurement requires that at least one full pulse is visible, and is most accurate if the pulse width is at least 0.4 divisions (20 pixels). Risetime and falltime measurements require that the edge being measured is visible, and will be most accurate if the transition takes at least 0.1 divisions (5 pixels).

It is best to have the power meter set on the fastest timebase possible that meets the edge visibility restrictions. Set the trace averaging as high as practical to reduce fluctuations and noise in the pulse timing measurements. Note that the timing of the edge transitions is defined by the settings of the SENSe:PULSe:DISTal, :MESIal and :PROXimal settings; see the descriptions for those commands. Units are the same as the channel's units.

Query Syntax READ<channel>:ARRay:AMEAsure:TIMe?
<channel> := {1 |2 |3 |4}

Example READ1:ARR:AMEA:TIM?

Query Response Returns: CC1, PulseFreq, CC2, PulsePeriod, CC3, PulseWidth, CC4, Offtime, CC5, DutyCycle, CC6, Risetime, CC7, Falltime, CC8, EdgeDly, CC9, Skew in Hz Where the CCn's are the measurement condition codes for each measurement.

Note:

Only available in **Pulse** mode.
Timebase must be set appropriately to allow measurements (see above)

10.4 READ:ARRay:CW:POWer?

Description Returns the current average, maximum, minimum powers or voltages and the peak-toaverage ratio of the specified channel. Units are the same as the channel's units. The peak-to-average ratio and marker ratio are returned in dB for logarithmic channel units, and percent for all other channel units. The values for maximum and minimum power will depend on the peak hold mode; see the description of the CALCulate:PKHLD command for details.

Query Syntax READ<channel>:ARRay:CW:POWer?
<channel> := {1 |2 |3 |4}

Example READ1:ARR:CW:POW?

Query Response Returns: CC1, Pavg, CC2, Pmax, CC3, Pmin, CC4, PkToAvgRatio Where the CCn's are the measurement condition codes for each measurement.

Note:

Available in **Pulse** and **Modulated** mode.

10.5 READ:ARRay:MARKer:POWer?

Description Returns an array of the current marker measurements for the specified channel. The array consists of the average, maximum, and minimum power and peak-to-average ratio between the two markers, powers at both markers, and the ratio of the two markers. The peak-to-average ratio and marker ratio are returned in dB for log units, and percent for linear units.

Query Syntax READ<channel>:ARRay:MARKer:POWer?
<channel> := {1 |2 |3 |4}

Example READ1:ARRay:MARKer:POWer?

Query Response Returns: CC1, Pavg, CC2, Pmax, CC3, Pmin, CC4, PkToAvgRatio, CC5, Pwr@Marker1, CC6, Pwr@Marker2, CC7, Mrk1/Mrk2 ratio Where the CCn's are the measurement condition codes for each measurement.

Note:

Available in **Pulse** and **Modulated** mode.

10.6 READ:CW:POWer?

Description Returns the current average amplitude reading in channel units.

Query Syntax READ<channel>:CW:POWer?
<channel> := {1 |2 |3 |4}

Example READ1:CW:POW?

Query Respond Returns: CC, Average power (watts, dBm) or average voltage (volts, dBv) Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode.

10.7 READ:INTERval:AVERage?

Description Returns the average power or voltage between marker 1 and marker 2 of the specified channel. The units will be the same as the specified channel.

Query Syntax READ<channel>:INTERval:AVERage?
<channel> := {1 |2 |3 |4}

Example READ1:INTER:AVER?

Query Respond Returns: CC, average power or voltage between markers Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode.

10.8 READ:INTERval:MAXFilt?

Description Returns the maximum filtered power or voltage in the time interval between marker 1 and marker 2 of the specified channel. The units will be the same as the specified channel.

Query Syntax READ<channel>:INTERval:MAXFilt?
<channel> := {1 |2 |3 |4}

Example READ1:INTER:MAXF?

Query Respond Returns: CC, maximum filtered power or voltage between the markers Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modlated** mode.

10.9 READ:INTERval:MINFilt?

Description For the specified channel, return the minimum filtered power or voltage in the time interval between marker 1 and marker 2. The units will be the same as the specified channel.

Queery Syntax READ<channel>:INTERval:MINFilt?
<channel> := {1 |2 |3 |4}

Example READ1:INTER:MINF?

Query Respond Returns: CC, minimum filtered power or voltage between markers Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode.

10.10 READ:INTERval:MAXimum?

Description For the specified channel, return the maximum instantaneous power or voltage between marker 1 and marker 2. The units will be the same as the specified channel.

Query Syntax READ<channel>:INTERval:AVERage?
<channel> := {1 |2 |3 |4}

Example READ1:INTER:AVER?

Query Respond Returns: CC, maximum instantaneous power or voltage between markers Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode.

10.11 READ:INTERval:MINimum?

Description Returns the minimum instantaneous power or voltage between marker 1 and marker 2 of the specified channel. The units will be the same as the specified channel.

Query Syntax READ<channel>:INTERval:MINimum?
<channel> := {1 |2 |3 |4}

Example READ1:INTER:MIN?

Query Respond Returns: CC, minimum instantaneous power or voltage between markers Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mde.

10.12 READ:INTERval:PKAVG?

Description Returns the peak-to-average ratio of the power or voltage between marker 1 and marker 2 of the specified channel. The units are dB for logarithmic channel units or percent for linear channel units.

Query Syntax READ<channel>:INTERval:PKAVG?
<channel> := {1 |2 |3 |4}

Example

Query Respond Returns: CC, peak-to-average ratio of power or voltage between markers Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode.

10.13 READ:MARKer:AVERage?

Description Returns the average power or voltage at the marker of the specified channel. The units are the same as the specified channel.

QuerySyntax READ<channel>:MARKer<marker>:AVERage?
<channel> := {1 |2 |3 |4}
<marker> := {1 |2}

Example READ1:MARK2:AVER?

Query Respond Returns: CC, average power or voltage at marker Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode.

10.14 READ:MARKer:DELTa?

Description Returns the difference between MK1 and MK2 of the specified channel. The units will be the same as marker units.

Query Syntax READ<channel>:MARKer:DELTa?
<channel> := {1 |2 |3 |4}

Example READ1:MARK:DELT?

Query Respond Returns: CC, (MK1 – MK2) Where CC is the measurement condition code.

10.15 READ:MARKer:MAXimum?

Description Returns the maximum power or voltage at the marker of the specified channel. The units are the same as the specified channel.

Query Syntax READ<channel>:MARKer<marker>:MAXimum?
<channel> := {1 |2 |3 |4}
<marker> := {1 |2}

Example READ1:MARK1:MAX?

Query Respond Returns: CC, maximum power or voltage at marker Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode.

10.16 READ:MARKer:MINimum?

Description Returns the minimum power or voltage at the marker of the specified channel. The units will be the same as the specified channel.

Query Syntax READ<channel>:MARKer<marker>:MINimum?
<channel> := {1 |2 |3 |4}
<marker> := {1 |2}

Example READ1:MARK2:MIN?

Query Respond Returns: CC, minimum power or voltage at marker Where CC is the measurement condition code.

Note:

Available in **Pulse** and **Modulated** mode.

10.17 READ:MARKer:CURsor:PERcent?

Description Returns the CCDF cursor y-axis position in percent with respect to the value set by MARKer:POSItion:POWer (CCDF cursor mode is set to Power Ref). If CCDF cursor mode is Percent, returns user setting. See MARKer:POSItion:POWer and MARKer:POSItion:PERcent. Refer to the PMX40 Instruction Manual Statistical Mode Automatic Measurements section for more information.

Query Syntax READ<channel>:MARKer:CURsor:PERcent?
<channel> := {1 |2 |3 |4}

Example READ1:MARK:CUR:PER?

Query Respond Returns: CC, percent CCDF Where CC is the measurement condition code.

Note:

Only available in **Statistical** mode.

10.18 READ:MARKer:CURsor:POWer?

Description Returns the CCDF cursor x-axis position in relative power with respect to the value set by MARKer:POSItion:PERcent (CCDF cursor mode is set to Percent). If CCDF cursor mode is Power Ref, returns user setting. See MARKer:POSItion:POWer and MARKer:POSItion:PERcent. Refer to the RF Power Meter Instruction Manual Statistical Mode Automatic Measurements section for more information.

Query Syntax READ<channel>:MARKer:CURsor:POWer?
<channel> := {1 |2 |3 |4}

Example READ1:MARK:CUR:POW?

Query Respond Returns: CC, relative power (dBr) CCDF Where CC is the measurement condition code.

Note:

Only available in **Statistical** mode.

10.19 READ:MARKer:RATio?

Description Returns the ratio of MK1 to MK2 of the specified channel. The units will be dB for logarithmic units or percent for linear units.

Query Syntax READ<channel>:Marker:RATio?
<channel> := {1 |2 |3 |4}

Example READ:Mark:RAT?

Query Respond Returns: CC, MK1/MK2 Where CC is the measurement condition code.

10.20 READ:MARKer:RDELTa?

Description For the specified channel return the difference between MK2 and MK1. The units will be the same as marker units.

Query Syntax READ<channel>:MARKer:RDELTa?
<channel> := {1 |2 |3 |4}

Example READ1:MARK:RDELT?

Query Respond Returns: CC, (MK2-MK1) Where CC is the measurement condition code.

10.21 READ:MARKer:RRATio?

Description For the specified channel return the ratio of MK2 to MK1. The units will be dB for logarithmic units or percent for linear units..

Query Syntax READ<channel>:MARKer:RRATio?
<channel> := {1 |2 |3 |4}

Example READ1:MARK:RRAT?

Query Respond Returns: CC, MK2/MK1 Where CC is the measurement condition code.

Sense Subsystem

The SENSe commands are used to configure the power meter for acquiring data. SENSe enables you to change measurement parameters such as filtering or averaging, sensor bandwidth, operating frequency and calfactors, and measurement gain or offset.

The numeric suffix of the SENSe program mnemonic in the SENSe commands refers to a hardware measurement "channel" that is SENSe1 and SENSe2 represent the instrument's SENSOR 1 and SENSOR 2 signal paths, respectively. The SENSe commands **DO NOT** affect the data processing and display portion of the measurement (see the CALCulate subsystem).

11.1 SENSe:AVERage

Description Set or return the number of traces averaged together to form the measurement result on the selected channel. Averaging can be used to reduce display noise on the visible trace, marker, and automatic pulse measurements. Trace averaging is a continuous process in which the measurement points from each sweep are weighted (multiplied) by an appropriate factor, and averaged into the existing trace data points. The most recent data will always have the greatest effect on the trace shape, and older measurements will be decayed at a rate determined by the averaging setting and trigger rate.

For timebase settings of 500 ns/div and faster, the rf power meter acquires samples using a technique called equivalent time or interleaved sampling. In this mode, not every pixel on the trace gets updated on each sweep, and the total number of sweeps needed to satisfy the AVERage setting will be increased by the sample interleave ratio of that particular timebase.

Command Syntax SENSe<channel>:AVERage <numeric value>
<channel> := {1 |2 |3 |4}
<numericvalue> := {1 to 16,384 (1 = no trace averaging)}

Query Syntax SENSe<channel>:AVERage?

Example SENS1:AVER 10,000 SENS1:AVER?

Query Response Returns: 10,000

Note:

Only available in **Pulse** mode.

11.2 SENSe:BANDwidth

Description Set or return the sensor video bandwidth for the selected sensor or the trigger channel bandwidth if a trigger channel is selected. HIGH is the normal setting for most measurements. The actual bandwidth is determined by the peak sensor model used. Use LOW bandwidth for additional noise reduction when measuring CW or signals with very low modulation bandwidth. If LOW bandwidth is used on signals with fast modulation, measurement errors will result because the sensor cannot track the fast changing envelope of the signal.

Command Syntax SENSe<channel>:BANDwidth <character data>
<channel> := {1 |2 |3 |4}
<character data> := {LOW |HIGH}

Query Syntax SENSe<channel>:BANDwidth?

Example SENSe1:BAND HIGH SENSe1:BAND?

Query Response Returns: HIGH

11.3 SENSe:CORRection:CALFactor

Description Set or return the frequency calfactor currently in use on the selected channel. Setting a calfactor with this command will override the "automatic" frequency calfactor that was calculated and applied when the operating frequency was set. Setting the operating frequency will override this calfactor setting.

Command Syntax SENSe<channel>:CORRection:CALFactor <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := {–3.00 to 3.00 dB}

Query Syntax SENSe<channel>:CORRection:CALFactor?

Example SENS1:CORR:CALF 1 SENS1:CORR:CALF?

Query Response Returns: 1 dB

11.4 SENSe:CORRection:DCYCle

Description Set or return the duty cycle correction factor currently in use on the selected channel.

Command Syntax SENSe<channel>:CORRection:DCYCle <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := {0.01 to 100.00 percent}

Query Syntax SENSe<channel>:CORRection:DCYCle?

Example SENS:CORR:DCYC 50
SENS:CORR:DCYC?

Query Response Returns: 50

Note:

Available with **CW Sensor** in **Modulated** mode.

11.5 SENSe:CORRection:FREQuency

Description Set or return the RF frequency for the current sensor. The appropriate frequency calfactor from the sensor's EEPROM table will be applied. Application of this calfactor cancels out the effect of variations in the flatness of the sensor's frequency response. If an explicit calfactor has been set, either manually or via the SENSe:CORRection:CALFactor command, entering a new frequency will override this calfactor and use only the "automatic" frequency calfactor.

Command Syntax SENSe<channel>:CORRection:FREQuency <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := {1e6 to 110.0e9 Hz}
Sensor may have narrower range. The range depends on channel bandwidth setting.

Query Syntax SENSe<channel>:CORRection:FREQuency?

Example SENS:CORR:FREQ 1e9 SENS:CORR:FREQ?

Query Response Returns: 1 GHz

11.6 SENSe:CORRection:OFFSet

Description Set or return a measurement offset in dB for the selected channel. This is used to compensate for external couplers, attenuators or amplifiers in the RF signal path ahead of the power sensor.

Command Syntax SENSe<channel>:CORRection:OFFSet <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := {–300.000 to 300.000 dB}

Query Syntax SENSe<channel>:CORRection:OFFSet?

Example SENS1:CORR:OFFS 1
SENS1:CORR:OFFS?

Query Response Returns: 1 dB

11.7 SENSe:FILTer:STATe

Description Set or return the current setting of the integration filter on the selected channel. OFF provides no filtering, and can be used at high signal levels when absolute minimum settling time is required. ON allows a user-specified integration time, from 2 milliseconds to 15 seconds (see SENSe:FILTer:TIMe command).

Setting the filter time will force the state to ON. AUTO uses a variable amount of filtering, which is set automatically by the power meter based on the current signal level to a value that gives a good compromise between measurement noise and settling time at most levels.

Command Syntax SENSe<channel>:FILTer:STATe <character data>

Query Syntax SENSe<channel>:FILTer:STATe?
<channel> := {1 |2 |3 |4}
<character data> := {OFF |ON |AUTO}

Example SENS1:FILT:STAT AUTO
SENS1:FILT:STAT?

Query Response Returns: AUTO

Note:

Only available in **Modulated** mode.

11.8 SENSe:FILTer:TIMe

Description Set or return the current length of the integration filter on the selected channel. If the filter state is set to AUTO, querying the time will return -0.01, and if set to OFF, a time query will return 0.00. Setting the filter time will force the state to ON.

Command Syntax SENSe<channel>:FILTer:TIMe <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := {0.002 to 16.000 seconds in 2 millisecond increments {

Query Syntax SENSe<channel>:FILTer:TIMe?

Example SENS1:FILT:TIM 1
SENS1:FILT:TIM?

Query Response Returns: 1 sec

Note:

Only available in **Modulated** mode.

11.9 SENSe:PULSe:DISTal

Description Set or return the pulse amplitude percentage, The pulse amplitude percentage defines the end of a rising edge or beginning of a falling edge transition. Typically, this is 90% voltage or 81% power relative to the "top level" of the pulse. This setting is used when making automatic pulse risetime and falltime calculations returned by READ:ARRay:AMEASure:POWer.

Command Syntax SENSe<channel>:PULSe:DISTal <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> = 50.00 to 100.00 percent

Query Syntax SENSe<channel>:PULSe:DISTal?

Example SENS1:PULS:DIST 50
SENS1:PULS:DIST?

Query Response Returns: 50 %

Note:

Only Available in **Pulse** mode.

11.10 SENSe:PULSe:ENDGT

Description Set or return the point on a pulse to define the end of the pulse's "active" interval. This point is defined in percent of the total pulse duration, with 0% corResponseing to the midpoint of the rising edge, and 100% corResponseing to the midpoint of the falling edge, as defined by the mesial setting. For most pulse "on" average power measurements, it is desirable to exclude the rising and falling intervals, and only measure power over the active portion of the pulse. This is often known as time gating, and is used for the automatic pulse measurements returned by READ:ARRay:AMEASure:POWer.

Command Syntax SENSe<channel>:PULSe:ENDGT <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := {60.00 to 100.00 percent}

Query Syntax SENSe<channel>:PULSe:ENDGT?

Example SENS:PULS:ENDGT 50
SENS:PULS:ENDGT?

Query Response Returns: 50 %

Note:

Only available in **Pulse** mode.

11.11 SENSe:PULSe:MESIal

Description Set or return the pulse amplitude percentage used to define the midpoint of a rising or falling edge transition. Typically, this is 50% voltage or 25% power relative to the "top level" of the pulse. This setting is used when making automatic pulse width and duty cycle calculations returned by FETCh:ARRay:AMEASure:POWer.

Command Syntax SENSe<channel>:PULSe:MESIal <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := {10.00 to 90.00 percent}

Query Syntax SENSe<channel>:PULSe:MESIa?

Example SENS:PULS:MESI 50
SENS:PULS:MESI

Query Response Returns: 50 %

Note:

Only available in **Pulse** mode.

11.12 SENSe:PULSe:PROXimal

Description Set or return the pulse amplitude percentage, which is used to define the beginning of a rising edge or end of a falling edge transition. Typically, this is 10% voltage or 1% power relative to the "top level" of the pulse. This setting is used when making automatic pulse risetime and falltime calculations returned by FETCh:ARRay:AMEASure:POWer.

Command Syntax SENSe<channel>:PULSe:PROXimal <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := {0.00 to 50.00 percent}

Query Syntax SENSe<channel>:PULSe:PROXimal?

Example
```
SENS1:PULS:PROX 25
SENS1:PULS:PROX?
```

Query Response Returns: 25 %

Note:

Only available in **Pulse** mode.

11.13 SENSe:PULSe:STARTGT

Description Set or return the point on a pulse used to define the beginning of the pulse's "active" interval. This point is defined in percent of the total pulse duration, with 0% corResponseing to the midpoint of the rising edge, and 100% corResponseing to the midpoint of the falling edge, as defined by the mesial setting. For most pulse "on" average power measurements, it is desirable to exclude the rising and falling intervals, and only measure power over the active portion of the pulse. This is often known as time gating, and is used for the automatic pulse measurements returned by FETCh:ARRay:AMEASure:POWer.

Syntax SENSe<channel>:PULSe:STARTGT <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := {0.00 to 40.00 percent}

Query Syntax SENSe<channel>:PULSe:STARTGT?

Example
```
SENS1:PULS:STARTGT 20
SENS1:PULS:STARTGT?
```

Query Response Returns: 20 %

Note:

Only available in **Pulse** mode.

11.14 SENSe:PULSe:UNIT

Description Set or return the units for entering the pulse distal, mesial and proximal levels. If the function is set to VOLTS, the pulse transition levels will be defined as the specified percentage in voltage. If set to WATTS, the levels are defined in percent power. Many pulse measurements call for 10% to 90% voltage (which equates to 1% to 81% power) for risetime and falltime measurements, and measure pulse widths from the half-power (–3dB, 50% power, or 71% voltage) points.

Syntax SENSe<channel>:PULSe:UNIT <character data>
<channel> := {1 |2 |3 |4}
<character data> := {WATTS, VOLTS}

Query Syntax SENSe<channel>:PULSe:UNIT?

Example SENS1:PULS:UNIT VOLTS
SENS1:PULS:UNIT?

Query Response Returns: VOLTS

Note:

Only available in **Pulse** mode.

11.15 SENSe:SENSOR:TYPE?

Description Returns the sensor type for the selected channel.

Query Syntax SENSe<channel>:SENSOR:TYPE?
<channel> := {1 |2 |3 |4}

Example SENS1:SENSOR:TYPE?

Query Response Returns: <return string data> := { NONE |CW |VOLT|PEAK}

System Subsystem

The system commands control system-level functions not directly related to instrument measurement performance. SYSTem commands are used to return error codes or messages from the power meter error queue, control hardware features (backlight and key beep), access the internal clock/calendar, and configure communication parameters for the GPIB and LAN interfaces.

12.1 SYSTem:BEEP[:ENABle]

Description Set or return the status of the audible keyboard beeper.

Command Syntax SYSTem:BEEP[:ENABle] <status>
<status> := {0 |1 or OFF |ON}

Query Syntax SYSTem:BEEP[:ENABle]?

Example SYST:BEEP ON
SYST:BEEP?

Query Response Returns: 1

12.2 SYSTem:COMMunicate:GPIB:ADDRess

Description Set or return the GPIB bus address.

Command Syntax SYSTem:COMMunicate:GPIB:ADDRess <numeric value>
<numeric value> := {1 to 30}

Query Syntax SYSTem:COMMunicate:GPIB:ADDRess?

Example SYST:COMM:GPIB:ADDR 3 SYST:COMM:GPIB:ADDR?

Query Response Returns: 3

12.3 SYSTem:COMMunicate:LAN:ADDRess

Description Set or return the IP address for the Ethernet port.

Syntax SYSTem:COMMunicate:LAN:ADDRess <character data>
<character data> := {instrument IP address in nnn.nnn.nnn.nnn (“dot decimal”) format}

Query Syntax SYSTem:COMMunicate:LAN:ADDRess?

Example SYST:COMM:LAN:ADDR 192.168.1.244 SYST:COMM:LAN:ADDR?

Query Response Returns: 192.168.1.244

Note:

DHCP/AutoIP must be disabled (OFF) to set the instrument IP address.

12.4 SYSTem:COMMunicate:LAN:DHCP

Description Set or return the state of DHCP/AutoIP system for the Ethernet port. If DHCP/AutoIP is enabled (1 |ON), the instrument will attempt to obtain its IP Address, Subnet Mask, and Default Gateway from a DHCP (dynamic host configuration protocol) server on the network. If no DHCP server is found, the instrument will select its own IP Address, Subnet Mask, and Default Gateway values using the "AutoIP" protocol. If DHCP/AutoIP is disabled (0 |OFF), the instrument will use the IP Address, Subnet Mask, and Default Gateway values that have been entered by the user.

Command Syntax SYSTem:COMMunicate:LAN:DHCP <state>
<state> = 0 |1 or OFF |ON

Query Syntax SYSTem:COMMunicate:LAN:DHCP?

Example SYST:COMM:LAN:DHCP ON
SYST:COMM:LAN:DHCP?

Query Response Returns: 1

12.5 SYSTem:COMMunicate:LAN:CURRent:ADDRess?

Description Returns the current IP address for the Ethernet port.

Query Syntax SYSTem:COMMunicate:LAN:CURRent:ADDRess?

Example SYST:COMM:LAN:CURR:ADDR?

Query Response Returns: <character data> := {current IP address in nnn.nnn.nnn.nnn ("dot decimal") format}

12.6 SYSTem:ERRor[:NEXT]?

Description Returns the next queued error code number followed by a quoted ASCII text string describing the error. Note that errors are stored in a "first-in-first-out" queue, so if more than one error has occurred, repeating this command will report the errors in the sequence they happened. The action of reading an error removes that error from the queue, so once the most recent error has been read, further queries will report a code of zero, and "No Error"

Query Syntax SYSTem:ERRor[:NEXT]?

Example SYST:ERR?

Query Response Returns: <numeric error code>, "QUOTED ERROR DESCRIPTION"
-200 Execution error

12.7 SYSTem:ERRor:CODE?

Description Returns the next queued error code number. Errors are stored in a "first-infirst-out" queue, so if more than one error has occurred, repeating this command will report the error codes in the sequence they happened. The action of reading an error removes that error from the queue, so once the most recent error has been read, any more queries will report a code of zero.

Query Syntax SYSTem:ERRor:CODE?

Example SYST:ERR:CODE?

Query Response Returns: <numeric error code>, "QUOTED ERROR DESCRIPTION"
-200 Execution error

12.8 SYSTem:ERRor:COUNt?

Description Returns the number of errors that currently exist in the error queue. A value of 0 means that there are no errors in the queue. Therefore, either no errors have occurred, or all errors have been read.

Query Syntax SYSTem:ERRor:COUNt?

Example SYST:ERR:COUN?

Query Response Returns: 1

12.9 SYSTem:PRESet

Description Set the meter's parameters to default. Equivalent to selecting "Initialize" from Measure > Meas. Settings on the rf power meter.

Command Syntax SYSTem:PRESet

Example SYST:PRES

12.10 SYSTem:VERSion?

Description Return the SCPI version compliance claimed.

Query Syntax SYSTem:VERSion?

Example SYST:VERS?

Query Response Returns: <character data> := {Version Code as <year.version> YYYY.V (will return 1999.0)}

Trace Subsystem

The TRACe commands are used to control the output of an acquired measurement array, which appears as a display trace when the power meter is in Graph mode. The TRACe commands allow outputting a channel's entire internal display trace (501 measurement points) as one large array, or selecting and returning the array in smaller portions. These commands are useful for capturing the displayed waveform and importing it into a database on the host.

13.1 TRACe [:AVERage]]:DATA[:NEXT]?

Description Return a delimited array of power or voltage pixel average values corresponding to all or a portion of the graph mode display trace for the selected channel. The Average trace is the data shown when envelope display mode is inactive. Note that the pixel values are returned without regard to display vertical scale and center settings. The array will consist of COUNT trace pixel values, beginning at pixel number INDEX, up to the last pixel of the trace (index = 500), and will be returned in channel units. The selected channel must be "ON" to return measurement data.

Query Syntax TRACe<channel>[:AVERage]]:DATA[:NEXT]?
<channel> := {1 |2 |3 |4}

Example TRACe<channel>:DATA?

Query Response Returns: P(index), P(index+1), P(index+2),...P(index+count) or V(index), V(index+1), V(index+2),...V(index+count)

13.2 TRACe:COUNt

Description Set or return the number of trace points, which will be returned each time the TRACe:DATA? query is issued. At the completion of each read, INDEX is automatically incremented by COUNT. If COUNT is set to a number greater than the number of points remaining in the trace, the array will be truncated. Setting COUNT to 501 (and INDEX to zero each time) will return the entire trace array.

Command Syntax TRACe<channel>:COUNt <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := {1 to 501}

Query Syntax TRACe<channel>:COUNt?

Example TRAC1:COUN 501

Query Response Returns: 501

13.3 TRACe:INDEX

Description Set or return the array index for the first trace point to be returned next time the TRACe:DATA? query is issued. Index 0 is the start of the trace buffer, and corresponds to the leftmost pixel on the graph display. Index 500 is the last point, and is the rightmost pixel. Each time a block of data is read, INDEX is automatically incremented by the COUNT value, so the full array can be split up into blocks of manageable size and read with successive TRACe:DATA? queries. INDEX must be reset to zero for each new trace that is to be dumped, whether or not all the points have been read.

Syntax TRACe<channel>:INDEX <numeric value>
<channel> := {1 |2 |3 |4}
<numeric value> := {0 to 500}

Query Syntax TRACe<channel>:INDEX?

Example TRAC1:INDEX 500
TRAC1:INDEX?

Query Response Returns: 500

Trigger Subsystem

The TRIGger commands are used to control synchronization of data acquisition with external events. TRIGger commands generally affect only Pulse Mode.

14.1 TRIGger:CDF:COUNt

Description Set or return the terminal count (sample population size) in millions of samples for Statistical Mode acquisition. When the terminal count is reached, the CCDF is considered "complete", and the instrument will halt acquisition if INITiate:CONTinuous is set to OFF. If INITiate:CONTinuous is ON, sample acquisition will continue in the manner specified by the TRIGger:CDF:DECImate setting.

CommandSyntax TRIGger:CDF:COUNt <numericvalue>
<numeric_value> := {1 to 4000 megasamples}

Query Syntax TRIGger:CDF:COUNt>

Example TRIG:CDF:COUN 100
TRIG:CDF:COUN?

Query Response Returns: 100

Note:

Only available in **Statistical** mode.

14.2 TRIGger:CDF:DECImate

Description Set or return the termination action when running continuously in Statistical Mode. This action occurs when the terminal count is reached (as defined by TRIGger:CDF:COUNt) or the terminal running time is reached (as defined by TRIGger:CDF:TIMe).

Command Syntax TRIGger:CDF:DECImate <action>
<action> := { DECIMATE, RESTART, STOP}

Query Syntax TRIGger:CDF:DECImate?

Example TRIG:CDF:DECI STOP
TRIG:CDF:DECI

Query Response Returns: STOP

Note:

Only available in **Statistical** mode.

14.3 TRIGger:CDF:TIMe

Description Set or return the terminal running time in seconds for Statistical Mode acquisition. When the terminal time is reached, the CCDF is considered "complete", and the instrument will halt acquisition if INITiate:CONTinuous is set to OFF. If INITiate:CONTinuous is ON, sample acquisition will continue in the manner specified by the TRIGger:CDF:DECImate setting.

Command Syntax TRIGger:CDF:TIMe <numeric value>
<numeric value> := {1 to 3600 seconds}

Query Syntax TRIGger:CDF:TIMe?

Example TRIG:CDF:TIM 10
TRIG:CDF:TIM?

Query Response Returns: 10 s

Note:

Only available in **Stattistical** mode.

14.4 TRIGger:DELay

Description Set or return the trigger delay time in seconds with respect to the trigger for the trigger display location in the LEFT position. Positive values cause the actual trigger to occur after the trigger condition is met. This places the trigger event to the left of the trigger point on the display, and is useful for viewing events during a pulse, some fixed delay time after the rising edge trigger. Negative trigger delay places the trigger event to the right of the trigger point on the display, and is useful for looking at events before the trigger edge.

Command Syntax TRIGger:DELay <numeric value>
Timebase setting Trigger Delay range (LEFT position only)
5 ns/div to 500 ns/div –4 ms to +100 ms
1 us/div to 10 ms/div +/- 4000 divisions
20 ms/div to 3600 sec/div –40 to +100 sec

Query Syntax TRIGger:DELay

Example TRIG:DEL 1 TRIG:DEL?

Query Response Returns: 1 s

Note:

Only available in **Pulse** mode.

14.5 TRIGger:HOLDoff

Description Set or return the trigger holdoff time in seconds. Trigger holdoff is used to disable the trigger for a specified amount of time after each trigger event. The holdoff time starts immediately after each valid trigger edge, and will not permit any new triggers until the time has expired. When the holdoff time is up, the trigger re-arms, and the next valid trigger event (edge) will cause a new sweep. This feature is used to help synchronize the power meter with burst waveforms such as a TDMA or GSM frame. The trigger holdoff resolution is 10 nanoseconds, and it should be set to a time that is just slightly shorter than the frame repetition interval.

Command Syntax TRIGger:HOLDoff <numeric value>
<numeric value> = {0.0 to 1.0 seconds in 10 ns increments |0.0 = no holdoff}

Query Syntax TRIGger:HOLDoff?

Example TRIG:HOLD 1 TRIG:HOLD?

Query Response Returns: 1

Note:

Only available in **Pulse** mode.

14.6 TRIGger:LEVel

Description Set or return the trigger level for synchronizing data acquisition with a pulsed input signal or external trigger pulses. The internal trigger level entered should include any global offset and will also be affected by the frequency cal factor. The available internal trigger level range is sensor dependent. For internal trigger sources, the trigger level is set and returned in dBm. The external trigger is set and returned in volts. Note that there is a small amount of hysteresis built into the trigger system, and the signal should have at least one dB greater swing in each direction past the trigger level setting, and somewhat more at low levels. Note that explicitly setting the trigger level while TRIGger:MODe is set to AUTOPKPK will cancel the AUTOPKPK setting, and force the trigger mode back to AUTO. In AUTOPKPK the Trigger Level menu will display "AUTO LEVEL".

Command Syntax TRIGger:LEVel <numeric value>
<numeric value> := {–40.0 to +20 dBm (plus offset, if any) (internal trigger sources)}
<numeric value> :={ +/-5.0 volts (external trigger) }

Query Syntax TRIGger:LEVel?

Example TRIG:LEVe 2
TRIG:LEVe?

Query Response Returns: 2 dB

Note:

Only available in **Pulse** mode.

14.7 TRIGger:MODe

Description Set or return the trigger mode for synchronizing data acquisition with pulsed signals. NORM mode will cause a sweep to be triggered each time the power level crosses the preset trigger level in the direction specified by TRIGger:SLOPe. If there are no edges that cross this level, no data acquisition will occur. AUTO mode operates in much the same way as NORM mode, but will automatically generate a trace if no trigger edges are detected for a period of time (100 to 500 milliseconds, depending on timebase). This will keep the trace updating even if the pulse edges stop. The AUTOPKPK mode operates the same as AUTO mode, but will adjust the trigger level to halfway between the highest and lowest power or voltage levels detected. This aids in maintaining synchronization with a pulse signal of varying level. Note that a setting of PKTOPK will be overridden and forced back to AUTO if a TRIGger:LEVel is set. The FREERUN mode force traces at a high rate to assist in locating the signal.

Command Syntax TRIGger:MODe <character data>
<character data> := {AUTO |AUTOPKPK |NORMAL |FREERUN}

Query Syntax TRIGger:MODe?

Example TRIG:MOD AUTO TRIG:MOD?

Query Response Returns: AUTO

Note:

Only available in **Pulse** mode.

14.8 TRIGger:POSition

Description Set or return the position of the trigger event on displayed sweep. Assuming zero trigger delay, setting the position to LEFT causes the entire trace to be post-trigger. Setting it to RIGHT causes the entire trace to be pre-trigger. And setting to MIDDLE will display both the pre- and post-trigger portions of the trace. Note that the TRIGger:DELay setting is in addition to this setting, and will cause the trigger position to appear in a different location. Setting the trigger position places the PMX40 in trigger position mode, which overrides the variable vernier settings of the TRIGger:VERNier command.

Command Syntax TRIGger:POSition <character data>
<character data> := {LEFT |MIDDLE |RIGHT}

Query Syntax TRIGger:POSition ?

Example TRIG:POS LEFT
TRIG:POS?

Query Response Returns: LEFT

Note:

Only available in **Pulse** mode.

TRIGger:SLOPe

Description Set or return the trigger slope or polarity. When set to POS, trigger events will be generated when a signal's rising edge crosses the trigger level threshold. When NEG, trigger events are generated on the falling edge of the pulse.

Command Syntax TRIGger:SLOPe <character data>
<character data> := {NEG |POS}

Query Syntax TRIGger:SLOPe?

Example TRIG:SLOP POS TRIG:SLOP?

Query Response Returns: POS

Note:

Only available in **Pulse** mode.

14.9 TRIGger:SOURce

Description Set or return the trigger source used for synchronizing data acquisition. The CH1 and CH2 (2 channel PMX40) settings use the signal from the associated sensor. EXT setting uses the signal applied to the rear panel TRIG IN connector.

Command Syntax TRIGger:SOURce <character data>
<character data> := {CH1 |CH2 |CH3 |CH4 |IND |EXT}

Query Syntax TRIGger:SOURce?

Example TRIG:SOUR EXT TRIG:SOUR?

Query Response Returns: EXT

Note:

Only available in **Pulse** mode.

14.10 TRIGger:VERNier

Description Set or return the fine position of the trigger event on displayed sweep. The position is given in divisions relative to the left edge of the screen, so with zero trigger delay, setting the vernier control to 0.0 causes the entire trace to be post-trigger. Setting it to 10.0 causes the entire trace to be pre-trigger. And setting to 5.0 will display both the pre- and post-trigger portions of the trace. Note that the TRIGger:DELay setting is in addition to this setting, and will cause the trigger position to appear in a different location. Setting the trigger vernier places the PMX40 in trigger vernier mode, which overrides the fixed "Left, Middle, Right" settings of the TRIGger:POSition command.

Command Syntax TRIGger:VERNier <numeric value>
<numeric value> := {–30.0 to 30.0 divisions}

Query Syntax TRIGger:VERNier?

Example TRIG:VERN 1 TRIG:VERN?

Query Response Returns: 1 dividsion

Note:

Only available in **Pulse** mode.
