RIGOL

RSA800 Series

Real-Time Spectrum Analyzer

Programming Guide

2026.06

Guaranty and Declaration

Copyright

Trademark Information

RIGOL® is the trademark of RIGOL TECHNOLOGIES CO., LTD.

Notices

- **RIGOL** products are covered by P.R.C. and foreign patents, issued and pending.
- **RIGOL** reserves the right to modify or change parts of or all the specifications and pricing policies at the company's sole decision.
- Information in this publication replaces all previously released materials.
- Information in this publication is subject to change without notice.
- **RIGOL** shall not be liable for either incidental or consequential losses in connection with the furnishing, use, or performance of this manual, as well as any information contained.
- Any part of this document is forbidden to be copied, photocopied, or rearranged without prior written approval of **RIGOL**.

Product Certification

RIGOL guarantees that this product conforms to the national and industrial standards in China as well as the ISO9001:2015 standard and the ISO14001:2015 standard. Others international standard conformance certifications are in progress

Contact Us

If you have any problem or requirement when using our products or this manual, please contact **RIGOL**.

E-mail: service@rigol.com

Website: *http://www.rigol.com*

Contents

1. Document Overview

This document is intended to guide users in programmatically controlling the RSA800 Series Spectrum Analyzers (GPSA and RTSA modes) with SCPI commands via remote interfaces.
The spectrum analyzer can communicate with a computer over USB and LAN interfaces.

Tip

For the latest version of this manual, download it from the official website of RIGOL (*http://www.rigol.com*).

Publication Number
PGD36100-1110

Software Version
00.01.00
Software upgrade might change or add product features. Please acquire the latest version of the manual from RIGOL website or contact RIGOL to upgrade the software.

Format Conventions in this Manual

1. **Key**

The front panel key is denoted by the menu key icon. For example, 

indicates the "System" key.

2. **Menu**

The menu item is denoted by the format of "Menu Name (Bold) + Character Shading" in the manual. For example, **Setup** indicates clicking or tapping the **Setup** sub-menu under the **System** menu to view the configuration items.

3. **Operation Procedures:**

The next step of the operation is denoted by ">" in the manual. For example,  > "**Save**" indicates that first clicking or tapping the icon  , then clicking or tapping "**Save**".

4. **Connector**

The connectors on the front or rear panel are usually denoted by the format of "Connector Name (Bold) + Square Brackets (Bold)". For example, **[TRIG IN]**.

Content Conventions in this Manual

The RSA800 series spectrum analyzer includes the following models. Unless otherwise specified, this manual takes RSA814 as an example to illustrate the functions and operation methods of RSA800 series spectrum analyzer.

Model	Frequency Range
RSA804	5 kHz to 4.5 GHz
RSA808	5 kHz to 8.5 GHz
RSA814	5 kHz to 14 GHz

2. Command System

This chapter introduces the syntax, function, parameters, and usage of each command for the RSA800 series spectrum analyzer.

Remarks:

1. The commands concerning the advanced measurement are only available for the RSA800 model installed with the relevant options. For details, refer to remarks for each command subsystem.
2. The commands related to the tracking generator are only valid in GPSA mode.
3. For the command set, unless otherwise specified, the query command returns "N/A" (without quotations in its return format) if no specified option is installed. If the queried function is disabled or improper type match is found, the query command will return "Error" (without quotations in its return format).
4. This manual takes RSA814 as an example to illustrate the range of the parameters in each command.

:ABORt Commands

:ABORt

Syntax
:ABORt

Description
Aborts the current measurement.

:CALCulate Commands

:CALCulate:BANDwidth|BWIDth:NDB

Syntax
:CALCulate:BANDwidth|BWIDth:NDB <rel_ampl>
:CALCulate:BANDwidth|BWIDth:NDB?

Description
Sets the N value in N dB BW measurement.
Queries the N value in N dB BW measurement.

Parameter

Name	Type	Range	Default
<rel_ampl>	Real	-140 dB to -0.01 dB	-3.01 dB

Return Format
The query returns the N value in scientific notation.

Example
The following command sets N to -4.
:CALCulate:BANDwidth:NDB -4
:CALCulate:BWIDth:NDB -4

The following query returns -4.000000e+00.
:CALCulate:BANDwidth:NDB?
:CALCulate:BWIDth:NDB?

:CALCulate:BANDwidth|BWIDth:RESult?

Syntax
:CALCulate:BANDwidth|BWIDth:RESult?

Description
Queries the measurement results of the N dB band.

Return Format
The query returns the bandwidth value in scientific notation.
If the two points which are located at both sides of the current marker with N dB fall or rise in peak amplitude are not found, the query returns -1.000000000e+02.

:CALCulate:BANDwidth|BWIDth:RLEFt?

Syntax
:CALCulate:BANDwidth|BWIDth:RLEFt?

Description
Queries the measurement value that is located at the left side of the current marker with N dB fall or rise in its peak amplitude.

Return Format
The query returns the mearment value in scientific notation.
If the frequency point that is located at the left side of the current marker with N dB fall or rise in its peak amplitude is not found, the query returns -1.000000000e+02.

:CALCulate:BANDwidth|BWIDth:RRIGht?

Syntax
:CALCulate:BANDwidth|BWIDth:RRIGht?

Description
Queries the measurement value that is located at the right side of the current marker with N dB fall or rise in its peak amplitude.

Return Format
The query returns the measurement value in scientific notation.

:CALCulate:BANDwidth|BWIDth[:STATe]

Syntax
:CALCulate:BANDwidth|BWIDth [:STATe] <bool>
:CALCulate:BANDwidth|BWIDth [:STATe]?

Description
Enables or disables the N dB bandwidth measurement function.
Queries the on/off status of the N dB bandwidth measurement function.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 0 or 1.

Example
The following command enables the N dB BW measurement function.
:CALCulate:BANDwidth:STATe ON or :CALCulate:BANDwidth:STATe 1
:CALCulate:BWIDth:STATe ON or :CALCulate:BWIDth:STATe 1

The following query returns 1.
:CALCulate:BANDwidth:STATe?
:CALCulate:BWIDth:STATe?

:CALCulate:LLINe:ALL:DELete

Syntax
:CALCulate:LLINe:ALL:DELete

Description
Deletes all the limit line data.

:CALCulate:LLINe:TEST[:STATe]

Syntax
:CALCulate:LLINe:TEST [:STATe] <bool>
:CALCulate:LLINe:TEST [:STATe]?

Description
Enables or disables the limit line test.
Queries the on/off state of the limit line test.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 0 or 1.

Example
The following command enables or disables the limit line test.
:CALCulate:LLINe:TEST:STATe ON or :CALCulate:LLINe:TEST:STATe 1

The following query command returns 1.
:CALCulate:LLINe:TEST:STATe?

:CALCulate:LLINe<n>:DISPlay:[:STATe]

Syntax
:CALCulate:LLINe<n>:DISPlay [:STATe] <bool>
:CALCulate:LLINe<n>:DISPlay [:STATe]?

Description
Enables or disables the display of the limit line.
Queries the display state of the limit line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 0 or 1.

Example
The following command enables the display of limit line1.
:CALCulate:LLINe1:DISPlay:STATe ON or :CALCulate:LLINe1:DISPlay:STATe 1

The following query command returns 1.
:CALCulate:LLINe1:DISPlay:STATe?

:CALCulate:LLINe<n>:TRACe

Syntax
:CALCulate:LLINe<n>:TRACe <real>
:CALCulate:LLINe<n>:TRACe?

Description
Sets the trace to be tested against the current limit line.
Queries the trace to be tested against the current limit line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	——
<real>	Integer	1|2|3|4|5|6	Refer to "Remarks"

Remarks
Limit Line 1 and 2: The default is Trace 1.
Limit Line 3 and 4: The default is Trace 2.

Limit Line 5 and 6: The default is Trace 3.

Return Format
The query returns 1, 2, 3, 4, 5, or 6.

Example
The following command sets that Trace 2 is tested against Limit Line 3.
:CALCulate:LLINe3:TRACe 2

The following query returns 2.
:CALCulate:LLINe3:TRACe?

:CALCulate:LLINe<n>:DELete

Syntax
:CALCulate:LLINe<n>:DELete

Description
Deletes the current limit line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1

Example
The following command deletes Limit Line 2.
:CALCulate:LLINe2:DELete

:CALCulate:LLINe<n>:DESCription

Syntax
:CALCulate:LLINe<n>:DESCription <string>
:CALCulate:LLINe<n>:DESCription?

Description
Sets the descriptions of the specified limit line.
Queries the descriptions of the specified limit line.

Parameter

Name	Type	Range	Default
<string>	ASCII String	——	--

Example
The following command sets the descriptions for Limit Line 1 to be RIGOL1998.
:CALCulate:LLINe1:DESCription RIGOL1998

The following query command returns RIGOL1998.
:CALCulate:LLINe1:DESCription?

:CALCulate:LLINe<n>:COMMent

Syntax
:CALCulate:LLINe<n>:COMMent <string>
:CALCulate:LLINe<n>:COMMent?
Description
Sets the comments for the specified limit line.
Queries the comments for the specified limit line.

Parameter

Name	Type	Range	Default
<string>	ASCII String	——	--

Example
The following command sets the comments for Limit Line 1 to "needs to be improved".
:CALCulate:LLINe1:COMMent needs to be improved

The following query command returns "needs to be improved".
:CALCulate:LLINe1:COMMent?

:CALCulate:LLINe<n>:MARGin

Syntax
:CALCulate:LLINe<n>:MARGin <rel_ampl>
:CALCulate:LLINe<n>:MARGin?

Description
Sets the margin for the selected limit line.
Queries the margin for the selected limit line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1
<rel_ampl>	Real	-40 dB to 0 dB	0 dB

Return Format
The query returns the margin for the limit line in scientific notation.

Example
The following command sets the margin for Limit Line 2 to be 1 dB.
:CALCulate:LLINe2:MARGin 1 dB

The following query returns 1.000000e+00.
:CALCulate:LLINe2:MARGin?

:CALCulate:LLINe<n>:MARGin:STATe

Syntax
:CALCulate:LLINe<n>:MARGin:STATe <bool>
:CALCulate:LLINe<n>:MARGin:STATe?

Description
Enables or disables the margin for the selected limit line.
Queries the setting state of the margin for the selected limit line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 0 or 1.

Example
The following command turns on the margin for Limit Line 2.
:CALCulate:LLINe2:MARGin:STATe ON or :CALCulate:LLINe2:MARGin:STATe 1

The following query returns 1.
:CALCulate:LLINe2:MARGin:STATe?

:CALCulate:LLINe<n>:OFFSet:Y

Syntax
:CALCulate:LLINe<n>:OFFSet:Y <real_ampl>
:CALCulate:LLINe<n>:OFFSet:Y?

Description
Set the Y offset of the specified limit line.
Queries the Y offset of the specified limit line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1
<real_ampl>	Real	0 dB to 40 dB	0 dB

Return Format
The query returns the offset Y in integer. The unit is dB.

Example
The following command sets the offset Y of Limit Line 2 to 10 dB.
:CALCulate:LLINe2:OFFSet:Y 10

The following query returns 10.
:CALCulate:LLINe2:OFFSet:Y?

:CALCulate:LLINe<n>:OFFSet:X

Syntax
:CALCulate:LLINe<n>:OFFSet:X <real_ampl>
:CALCulate:LLINe<n>:OFFSet:X?

Description
Set the X offset of the specified limit line.
Queries the X offset of the specified limit line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1
<real_ampl>	Real	-50 Hz to 50 Hz	0 Hz

Return Format
The query returns the X offset in integer. The unit is Hz.

Example
The following command sets the X offset of Limit Line 2 to 10 Hz.
:CALCulate:LLINe2:OFFSet:Y 10
The following query returns 10.
:CALCulate:LLINe2:OFFSet:Y?

:CALCulate:LLINe<n>:OFFSet:UPDate

Syntax
:CALCulate:LLINe<n>:OFFSet:UPDate

Description
Applies the offset to the specified limit line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1

Return Format
N/A

Example
The following command applies the offset to Limit Line 1.
:CALCulate:LLINe1:OFFSet:UPDate

:CALCulate:LLINe<n>:FREQuency:CMODe:RELative

Syntax
:CALCulate:LLINe<n>:FREQuency:CMODe:RELative <bool>
:CALCulate:LLINe<n>:FREQuency:CMODe:RELative?

Description
Enables or disables the coupling functions between the selected limit line data points and the center frequency.
Queries the setting state of the coupling functions between the selected limit line data points and the center frequency.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
OFF|0: disables the coupling function. That is, "Fixed" is selected under **Y To RF**, and the frequency of the current limit line editing point is not affected by the center frequency.
ON|1: enables the coupling function. That is, "Relative" is selected under **X To CF**, and the frequency of the current limit line editing point is the difference between the frequency of the point and that of the current center frequency. At this time, if the center frequency changes, then the position of the current editing point will move left and right with it.

Return Format
The query returns 0 or 1.

Example
The following command disables the coupling functions between data points of Limit Line 2 and the center frequency.
:CALCulate:LLINe2:FREQuency:CMODe:RELative OFF
or :CALCulate:LLINe2:FREQuency:CMODe:RELative 0

The following query returns 0.
:CALCulate:LLINe2:FREQuency:CMODe:RELative?

:CALCulate:LLINe<n>:AMPLitude:CMODe:RELative

Syntax
:CALCulate:LLINe<n>:AMPLitude:CMODe:RELative <bool>
:CALCulate:LLINe<n>:AMPLitude:CMODe:RELative?

Description
Enables or disables the coupling functions between the selected limit line data points and the reference level.
Queries the setting state of the coupling functions between the selected limit line data points and the reference level.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
OFF|0: disables the coupling function. That is, "Fixed" is selected under **X Axis Type**, and the amplitude of the current limit line editing point is not affected by the reference level.
ON|1: enables the coupling function. That is, "Relative" is selected under **X Axis Type**, and the amplitude of the current editing point is the difference between the amplitude of the point and that of the current reference level. At this time, if the reference level changes, then the position of the current editing point changes along with the reference level.

Return Format
The query returns 0 or 1.

Example
The following command disables the coupling functions between the selected limit line data points and the reference level.
:CALCulate:LLINe2:AMPLitude:CMODe:RELative OFF or :CALCulate:LLINe2:AMPLitude:CMODe:RELative 0

The following query returns 0.
:CALCulate:LLINe2:AMPLitude:CMODe:RELative?

:CALCulate:LLINe<n>:AMPLitude:INTerpolate:TYPE

Syntax
:CALCulate:LLINe<n>:AMPLitude:INTerpolate:TYPE <type>
:CALCulate:LLINe<n>:AMPLitude:INTerpolate:TYPE?

Description
Sets the amplitude interpolation type of the specified limit line.
Queries the amplitude interpolation type of the specified limit line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1
<type>	Discrete	LINear|LOG	LINear

Remarks
LINear: linear type.
LOG: log type.
Return Format
The query returns LIN or LOG.

Example
The following command sets the amplitude interpolation type of the Limit Line 2 to LOG.
:CALCulate:LLINe2:AMPLitude:INTerpolate:TYPE LOG

The following query returns LOG.
:CALCulate:LLINe2:AMPLitude:INTerpolate:TYPE?

:CALCulate:LLINe<n>:CONTrol:INTerpolate:TYPE

Syntax
:CALCulate:LLINe<n>:CONTrol:INTerpolate:TYPE <type>
:CALCulate:LLINe<n>:CONTrol:INTerpolate:TYPE?

Description
Sets the frequency interpolation type of the specified limit line.
Queries the frequency interpolation type of the specified limit line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1
<type>	Discrete	LINear|LOG	LOG

Remarks
LINear: linear type.
LOG: log type.

Return Format
The query returns LIN or LOG.

Example
The following command sets the frequency interpolation type of Limit Line 2 to LINear.
:CALCulate:LLINe2:CONTrol:INTerpolate:TYPE LINear

The following query returns LIN.
:CALCulate:LLINe2:CONTrol:INTerpolate:TYPE?

:CALCulate:LLINe<n>:TYPE

Syntax
:CALCulate:LLINe<n>:TYPE <enum>
:CALCulate:LLINe<n>:TYPE?

Description
Sets the type of the specified limit line.
Queries the type of the specified limit line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1
<enum>	Discrete	UPPer|LOWer	Refer to "Remarks"

Remarks
Limit Line 1, Limit Line 3, and Limit Line 5 belong to the upper type; Limit Line 2, Limit Line 4, and Limit Line 6

belong to the lower type.

Return Format
The query returns UPP or LOW.

Example
The following command sets Limit Line 2 to be the upper type.
:CALCulate:LLINe2:TYPE UPPer

The following query returns UPP.
:CALCulate:LLINe2:TYPE?

:CALCulate:LLINe<n>:BUILd

Syntax
:CALCulate:LLINe<n>:BUILd <trace>

Description
Builds the limit line from the selected trace.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1
<trace>	Discrete	TRACe1|TRACe2|TRACe3|TRACe4|TRACe5|TRACe6	——

Example
The following command builds Limit Line 2 from Trace2.
:CALCulate:LLINe2:BUILd TRACe2

:CALCulate:LLINe<n>:COPY

Syntax
:CALCulate:LLINe<n>:COPY <copy>

Description
Copies the selected limit line to the current limit line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1
<copy>	Discrete	LLINe1|LLINe2|LLINe3|LLINe4|LLINe5|LLINe6	——

Remarks
If the limit line to be copied that you select is the same as the current limit line, no operation should be performed.

Example
The following command copies Limit Line 1 to Limit Line 2.
:CALCulate:LLINe2:COPY LLINe1

:CALCulate:LLINe<n>:DATA

Syntax

:CALCulate:LLINe<n>:DATA <x>,<ampl>,<connect>{,<x>,<ampl>,<connect>}
:CALCulate:LLINe<n>:DATA?

Description
Edits one limit line, and marks it with n.
Queries the limit line data that you are editing currently.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1
<x>	Real	0 Hz to 6.5 GHz (X-axis indicates frequency) 0 us to 6.5 ks (X-axis indicates time)	——
<ampl>	Real	-1000 dBm to 1000 dBm	——
<connect>	Discrete	0|1	0

Remarks
<x>: indicates frequency or time.
<apml>: indicates the amplitude. By default, its unit is dBm. The same X value can be configured with at most two amplitude values.
<connect>: can be configured with 0 or 1. When it is configured with 1, it indicates that the current point connects with the previous point to determine the limit line; when configured with 0, it indicates that the current point is disconnected from the previous point. The <connect> value of the first point can be configured with 0.

Return Format
Queries the returned limit line data that you are editing currently.

Example
The following command edits a limit line that contains three points, and marks it Limit Line 2.
:CALCulate:LLINe2:DATA 50,100,0,100,150,1,200,200,1

The following query returns 50,100.000000,0,100,150.000000,1,200,200.000000,1.
:CALCulate:LLINe2:DATA?

:CALCulate:MARKer:AOFF

Syntax
:CALCulate:MARKer:AOFF

Description
Turns off all the enabled markers.

:CALCulate:MARKer:PEAK: MPEaks

Syntax
:CALCulate:MARKer:PEAK: MPEaks <num>
:CALCulate:MARKer:PEAK: MPEaks?

Description
Set the number of peaks.
Queries the number of peaks.

Parameter

Name	Type	Range	Default
<num>	Integer	-	20

Return Format
The query returns the number of peaks in integer.

Example
The following command sets the number of peaks to 12.
:CALCulate:MARKer:PEAK: MPEaks 12

The following query returns 12.
:CALCulate:MARKer:PEAK: MPEaks?

:CALCulate:MARKer:COUPle[:STATe]

Syntax
:CALCulate:MARKer:COUPle[:STATe] <bool>
:CALCulate:MARKer:COUPle[:STATe]?

Description
Sets the marker coupling state.
Queries the on/off state of the marker coupling.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
When you enable the couple marker function, moving any marker will make other markers (that are not fixed or off) move along with it.

Return Format
The query returns 0 or 1.

Example
The following command disables the couple marker function.
:CALCulate:MARKer:COUPle:STATe OFF or :CALCulate:MARKer:COUPle:STATe 0

The following query returns 0.
:CALCulate:MARKer:COUPle:STATe?

:CALCulate:MARKer:PEAK:EXCursion

Syntax
:CALCulate:MARKer:PEAK:EXCursion < ampl>
:CALCulate:MARKer:PEAK:EXCursion?

Description
Sets the excursion of the peak. Its unit is dB.
Queries the excursion of the peak.

Parameter

Name	Type	Range	Default
<ampl>	Real	0 dB to 100 dB	6 dB

Return Format
The query returns the peak excursion in scientific notation.

Example
The following command sets the peak excursion to 12 dB.
:CALCulate:MARKer:PEAK:EXCursion 12

The following query returns 1.200000e+01.
:CALCulate:MARKer:PEAK:EXCursion?

:CALCulate:MARKer:PEAK:EXCursion:STATe

Syntax
:CALCulate:MARKer:PEAK:EXCursion:STATe <bool>
:CALCulate:MARKer:PEAK:EXCursion:STATe?

Description
Enables or disables the peak excursion.
Queries the on/off state of the peak excursion.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Return Format
The query returns 0 or 1.

Example
The following command enables the peak excursion function.
:CALCulate:MARKer:PEAK:EXCursion:STATe ON or :CALCulate:MARKer:PEAK:EXCursion:STATe 1

The following query returns 1.
:CALCulate:MARKer:PEAK:EXCursion:STATe?

:CALCulate:MARKer:PEAK:SEARch:MODE

Syntax
:CALCulate:MARKer:PEAK:SEARch:MODE <mode>
:CALCulate:MARKer:PEAK:SEARch:MODE?

Description
Sets the peak search mode.
Queries the peak search mode.

Parameter

Name	Type	Range	Default
<mode>	Discrete	PARameter|MAXimum	MAXimum

Remarks
PARameter: indicates parameter. If "parameter" is selected under search mode, the system will search for the peak of the specified parameter on the trace.
MAXimum: indicates maximum. If Max is selected under Search Mode, the system will search for the maximum value on the trace.
The command is only valid for the peak search executed by sending the *:CALCulate:MARKer<n>:MAXimum* command. Other items under the peak search menu such as Next Peak, Next Peak Right, Next Peak Left, and Minimum Peak are all searched based on "Config" (parameter).
Return Format
The query returns PAR or MAX.

Example
The following command sets the peak search mode to be "Para".
:CALCulate:MARKer:PEAK:SEARch:MODE PARameter

The following query returns PAR.
:CALCulate:MARKer:PEAK:SEARch:MODE?

:CALCulate:MARKer:PEAK:SORT

Syntax
:CALCulate:MARKer:PEAK:SORT <type>
:CALCulate:MARKer:PEAK:SORT?

Description
Sets the sorting order of the data displayed in the peak table.
Queries the sorting order of the data displayed in the peak table.

Parameter

Name	Type	Range	Default
<type>	Discrete	FREQuency|AMPLitude	AMPLitude

Remarks
FREQuency: lists the peaks in order of ascending frequency.
AMPLitude: lists the peaks in order of descending amplitude.

Return Format
The query returns FREQ or AMPL.

Example
The following command sets the data in the peak table to be sorted in descending amplitude.
:CALCulate:MARKer:PEAK:SORT AMPLitude

The following query returns AMPL.
:CALCulate:MARKer:PEAK:SORT?

:CALCulate:MARKer:PEAK:TABLe:READout

Syntax
:CALCulate:MARKer:PEAK:TABLe:READout <type>
:CALCulate:MARKer:PEAK:TABLe:READout?

Description
Sets the peak table readout.
Queries the peak table readout.

Parameter

Name	Type	Range	Default
<type>	Discrete	ALL|GTDLine|LTDLine	ALL

Remarks
ALL: lists all the peaks defined by the peak criteria, and displays data based on the current sorting order in the peak table.
GTDLine: lists the peaks that are above the defined display line and that meet the peak criteria.
LTDLine: lists the peaks that are below the defined display line and that meet the peak criteria.

Return Format
The query returns ALL, GTDL, or LTDL.

Example
The following command lists all the peaks that are above the defined display line and that meet the peak criteria.
:CALCulate:MARKer:PEAK:TABLe:READout GTDLine

The following query returns GTDL.
:CALCulate:MARKer:PEAK:TABLe:READout?

:CALCulate:MARKer:PEAK:TABLe[:STATe]

Syntax
:CALCulate:MARKer:PEAK:TABLe[:STATe] <bool>
:CALCulate:MARKer:PEAK:TABLe[:STATe]?

Description
Enables or disables the peak table.
Queries the on/off status of the peak table.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 0 or 1.

Example
The following command enables the peak table.
:CALCulate:MARKer:PEAK:TABLe:STATe ON or :CALCulate:MARKer:PEAK:TABLe:STATe 1

The following query returns 1.
:CALCulate:MARKer:PEAK:TABLe:STATe?

:CALCulate:MARKer:PEAK:THReshold

Syntax
:CALCulate:MARKer:PEAK:THReshold <ampl>
:CALCulate:MARKer:PEAK:THReshold?

Description
Sets the peak threshold. Its default unit is dBm.
Queries the peak threshold.

Parameter

Name	Type	Range	Default
<ampl>	Real	-200 dBm to 0 dBm	-90 dBm

Return Format
The query returns the peak threshold in scientific notation.

Example
The following command sets the peak threshold to -100 dB.
:CALCulate:MARKer:PEAK:THReshold -100

The following query returns -1.000000e+02.
:CALCulate:MARKer:PEAK:THReshold?

:CALCulate:MARKer:PEAK:THReshold:STATe

Syntax
:CALCulate:MARKer:PEAK:THReshold:STATe <bool>
:CALCulate:MARKer:PEAK:THReshold:STATe?

Description
Sets the peak threshold state.
Queries the peak threshold state.
Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Return Format
The query returns 0 or 1.

Example
The following command enables the peak threshold function.
:CALCulate:MARKer:PEAK:THReshold:STATe ON or :CALCulate:MARKer:PEAK:THReshold:STATe 1

The following query returns 1.
:CALCulate:MARKer:PEAK:THReshold:STATe?

:CALCulate:MARKer:PEAK:THReshold:LINE:STATe

Syntax
:CALCulate:MARKer:PEAK:THReshold:LINE:STATe <bool>
:CALCulate:MARKer:PEAK:THReshold:LINE:STATe?

Description
Sets the status of the threshold line.
Queries the status of the threshold line.
Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Return Format
The query returns 0 or 1.

Example
The following command enables the peak threshold line.
:CALCulate:MARKer:PEAK:THReshold:LINE:STATe ON
or :CALCulate:MARKer:PEAK:THReshold:LINE:STATe 1

The following query returns 1.
:CALCulate:MARKer:PEAK:THReshold:LINE:STATe?

:CALCulate:MARKer:PEAK:MPEaks?

Syntax
:CALCulate:MARKer:PEAK:MPEaks?

Description
Queries the number of peaks.

:CALCulate:MARKer:NEXT

Syntax
:CALCulate:MARKer:NEXT

Description
Moves to the next marker.

:CALCulate:MARKer:TABLe[:STATe]

Syntax
:CALCulate:MARKer:TABLe[:STATe] <bool>
:CALCulate:MARKer:TABLe[:STATe]?

Description
Sets the marker table state.
Queries the on/off state of the marker table.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 0 or 1.

Example
The following command disables the marker table.
:CALCulate:MARKer:TABLe:STATe OFF or :CALCulate:MARKer:TABLe:STATe 0

The following query returns 0.
:CALCulate:MARKer:TABLe:STATe?

:CALCulate:MARKer:TRCKing[:STATe]

Syntax
:CALCulate:MARKer:TRCKing[:STATe] <bool>
:CALCulate:MARKer:TRCKing[:STATe]?

Description
Sets the status of the signal track.
Queries the on/off status of the signal track.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
When Signal Track is enabled, the instrument will execute a peak search after each sweep, and set the frequency value at the current peak to be the center frequency to keep the signal always displayed at the center of the screen.
The commands are valid only in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the signal track.
:CALCulate:MARKer:TRCKing:STATe ON or :CALCulate:MARKer:TRCKing:STATe 1

The following query returns 1.
:CALCulate:MARKer:TRCKing:STATe?

:CALCulate:MARKer<n>:MODE

Syntax
:CALCulate:MARKer<n>:MODE <mode>
:CALCulate:MARKer<n>:MODE?

Description
Sets the type of the specified marker.
Queries the type of the specified marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<mode>	Discrete	POSition|DELTa|FIXed|OFF	OFF

Remarks
POSition: indicates the normal marker.
DELTa: indicates difference between two data points.
FIXed: indicates that the marker is fixed.
OFF: turns off the selected marker.
The PvT only supports setting the marker on Trace 1.

Return Format
The query returns POS, DELT, FIX, or OFF.

Example
The following command sets the type of Marker 1 to Position.
:CALCulate:MARKer1:MODE POSition

The following query returns POS.
:CALCulate:MARKer1:MODE?

:CALCulate:MARKer<n>:PTPeak

Syntax
:CALCulate:MARKer<n>:PTPeak

Description
Performs the Pk-Pk search.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

Remarks

After the command is executed, the marker mode of the specified marker selects "Delta" automatically. The peak search results will be marked by the reference marker (by default, the next marker), and the minimum search will be marked by the Delta marker.

Example
The following command performs the peak-peak search, and marks the peak-peak position with the reference marker (Marker 2) and the Delta marker (Marker 1), respectively.
:CALCulate:MARKer1:PTPeak

:CALCulate:MARKer<n>:REFerence

Syntax
:CALCulate:MARKer<n>:REFerence <integer>
:CALCulate:MARKer<n>:REFerence?

Description
Sets the reference marker for the specified marker.
Queries the reference marker for the specified marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<integer>	Integer	1 to 8	By default, the reference marker is the marker next to it.

Remarks
Each marker can have another marker to be its reference marker.
It the current marker is a Delta marker, the measurement result of the marker will be determined by the reference marker.
Any marker cannot have itself to be the reference marker.
The PvT only supports setting the marker on Trace 1.

Example
The following command sets the reference marker for Marker 1 to 2.
:CALCulate:MARKer1:REFerence 2

The following query returns 2.
:CALCulate:MARKer1:REFerence?

:CALCulate:MARKer<n>:LINes[:STATe]

Syntax
:CALCulate:MARKer<n>:LINes[:STATe] <bool>
:CALCulate:MARKer<n>:LINes[:STATe]?

Description
Enables or disables the marker line of the specified marker.
Queries the setting state for the marker line of the specified marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
If the marker is not visible in the selected area, enable the marker line function to extend the marker line to the

display area for better observation.
The PvT only supports setting the marker on Trace 1.

Return Format
The query returns 0 or 1.

Example
The following command enables the marker line of Marker 1.
:CALCulate:MARKer1:LINes:STATe ON or :CALCulate:MARKer1:LINes:STATe 1

The following query returns 1.
:CALCulate:MARKer1:LINes:STATe?

:CALCulate:MARKer<n>:CPSearch[:STATe]

Syntax
:CALCulate:MARKer<n>:CPSearch[:STATe] <bool>
:CALCulate:MARKer<n>:CPSearch[:STATe]?

Description
Enables or disables the continuous peak search.
Queries the on/off status of continuous peak search.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 0 or 1.

Example
The following command performs the continuous peak search function, and marks with Marker 1.
:CALCulate:MARKer1:CPSearch:STATe ON or :CALCulate:MARKer1:CPSearch:STATe 1

The following query returns 1.
:CALCulate:MARKer1:CPSearch:STATe?

:CALCulate:MARKer<n>:CMSearch[:STATe]

Syntax
:CALCulate:MARKer<n>:CMSearch[:STATe] <bool>
:CALCulate:MARKer<n>:CMSearch[:STATe]?

Description
Sets the on/off status of continuous Min search.
Queries the on/off status of continuous Min search.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 0 or 1.

Example
The following command enables the continuous Min search function, and marks with Marker 1.
:CALCulate:MARKer1:CPSearch:STATe ON or :CALCulate:MARKer1:CPSearch:STATe 1

The following query returns 1.
:CALCulate:MARKer1:CMSearch:STATe?

:CALCulate:MARKer<n>:FCOunt:GATetime

Syntax
:CALCulate:MARKer<n>:FCOunt:GATetime <time>
:CALCulate:MARKer<n>:FCOunt:GATetime?

Description
Sets the gate time for the specified marker. The unit is s.
Queries the gate time for the specified marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<time>	Real	1 us to 500 ms	100 ms

Remarks
This command is only available for GPSA mode.

Return Format
The query returns the gate time for the specified marker in scientific notation.

Example
The following command sets the gate time for Marker 2 to 10 ms.
:CALCulate:MARKer2:FCOunt:GATetime 0.01

The following query returns 1.000000e-02.
:CALCulate:MARKer2:FCOunt:GATetime?

:CALCulate:MARKer<n>:FCOunt:GATetime:AUTO

Syntax
:CALCulate:MARKer<n>:FCOunt:GATetime:AUTO <bool>
:CALCulate:MARKer<n>:FCOunt:GATetime:AUTO?

Description
Turns on or off the auto gate time for the specified marker.
Queries the on/off status of the auto gate time for the specified marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only available for GPSA mode.

Return Format
The query returns 1 or 0.

Example
The following command enables the auto gate time for Marker 2.
:CALCulate:MARKer2:FCOunt:GATetime:AUTO ON or :CALCulate:MARKer2:FCOunt:GATetime:AUTO 1

The following query returns 1.
:CALCulate:MARKer2:FCOunt:GATetime:AUTO?

:CALCulate:MARKer<n>:FCOunt[:STATe]

Syntax
:CALCulate:MARKer<n>:FCOunt[:STATe] <bool>
:CALCulate:MARKer<n>:FCOunt[:STATe]?

Description
Enables or disables the marker counter.
Queries the on/off status of the marker counter.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only available for GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the frequency counter function of Marker 2.
:CALCulate:MARKer2:FCOunt:STATe ON or :CALCulate:MARKer2:FCOunt:STATe 1

The following query returns 1.
:CALCulate:MARKer2:FCOunt:STATe?

:CALCulate:MARKer<n>:FCOunt:X?

Syntax
:CALCulate:MARKer<n>:FCOunt:X?

Description
Queries the frequency count X of the specified marker counter.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

Remarks
This command is only available for GPSA mode.

Return Format
The query returns the readout in integer. Its unit is Hz.
When the frequency counter function is disabled, the command returns NAN.

:CALCulate:MARKer<n>:FUNCtion

Syntax
:CALCulate:MARKer<n>:FUNCtion <func>
:CALCulate:MARKer<n>:FUNCtion?

Description
Sets the band function type.
Queries the band function type.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<func>	Discrete	NOISe|BPOWer|BDENsity|OFF	OFF

Remarks
NOISe: indicates the noise measurement.
BPOWer: indicates the band power.
BDENsity: indicates the band density.
OFF: disables the band function.

Return Format
The query returns NOIS, BPOW, BDEN, or OFF.

Example
The following command sets the measurement type of Marker 1 to noise measurement.
:CALCulate:MARKer1:FUNCtion NOISe

The following query returns NOIS.
:CALCulate:MARKer1:FUNCtion?

:CALCulate:MARKer<n>:FUNCtion:BAND:LEFT

Syntax
:CALCulate:MARKer<n>:FUNCtion:BAND:LEFT <freq>
:CALCulate:MARKer<n>:FUNCtion:BAND:LEFT?

Description
Sets the left edge frequency or time for the band of the selected marker.
Queries the left edge frequency or time for the band of the selected marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	——
<freq>	Real	0 to band right	center frequency-bandwidth/2 acq time/2 - bandwidth/2 (PvT) sweep time/2 - bandwidth/2 (zero span)

Remarks
This command is only valid when the corresponding bandwidth function is enabled.
This command can be used to set the left edge frequency or time for the band of the selected marker in zero span mode in GPSA mode or in PvT view in RTSA mode.

Return Format
The query returns the left edge frequency or time for the band of the selected marker.

Example
The following command sets the left edge frequency or time for the band of Marker 1 to 2 MHz.
:CALCulate:MARKer1:FUNCtion:BAND:LEFT 2000000

The following query returns 2.000000000e+06.
:CALCulate:MARKer1:FUNCtion:BAND:LEFT?

:CALCulate:MARKer<n>:FUNCtion:BAND:RIGHt

Syntax
:CALCulate:MARKer<n>:FUNCtion:BAND:RIGHt <freq>
:CALCulate:MARKer<n>:FUNCtion:BAND:RIGHt?

Description
Sets the right edge frequency or time for the band of the selected marker.
Queries the right edge frequency or time for the band of the selected marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<freq>	Real	band left to +∞	center frequency+bandwidth/2 acq time/2 + bandwidth/2 (PvT) sweep time/2 + bandwidth/2 (zero span)

Remarks
This command is only valid when the corresponding bandwidth function is enabled.
This command can be used to set the right edge frequency or time for the band of the selected marker in zero span mode in GPSA mode or in PvT view in RTSA mode.

Return Format
The query returns the right edge frequency or time for the band of the selected marker in scientific notation.

Example
The following command sets the right edge frequency for the band of Marker 1 to 4 GHz.
:CALCulate:MARKer1:FUNCtion:BAND:RIGHt 4000000000

The following query returns 4.000000000e+09.
:CALCulate:MARKer1:FUNCtion:BAND:RIGHt?

:CALCulate:MARKer<n>:FUNCtion:BAND:SPAN

Syntax
:CALCulate:MARKer<n>:FUNCtion:BAND:SPAN <freq>
:CALCulate:MARKer<n>:FUNCtion:BAND:SPAN?

Description
Sets the band span of the specified marker.
Queries the band span of the specified marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<freq>	Real	0 to +∞	span/20 acq time/20 (PvT)

			sweep time/20 (zero span)

Remarks
This command is only valid when the corresponding bandwidth function is enabled.
This command can be used to set the band span of the specified marker in zero span mode in GPSA mode or in PvT view in RTSA mode.

Return Format
The query returns the band span involved in the calculation for the band function in scientific notation.

Example
The following command sets the band span of Marker 1 to 500 MHz.
:CALCulate:MARKer1:FUNCtion:BAND:SPAN 500000000

The following query returns 5.000000000e+08.
:CALCulate:MARKer1:FUNCtion:BAND:SPAN?

:CALCulate:MARKer<n>:FUNCtion:BAND:SPAN:AUTO

Syntax
:CALCulate:MARKer<n>:FUNCtion:BAND:SPAN:AUTO <bool>
:CALCulate:MARKer<n>:FUNCtion:BAND:SPAN:AUTO?

Description
Sets the band span auto state of the specified marker.
Queries the band span auto state of the specified marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
This command is only valid when the corresponding bandwidth function is enabled.

Return Format
The query returns 1 or 0.

Example
The following command enables the band span auto function of Marker 1.
:CALCulate:MARKer1:FUNCtion:BAND:SPAN:AUTO ON
or :CALCulate:MARKer1:FUNCtion:BAND:SPAN:AUTO 1

The following query returns 1.
:CALCulate:MARKer1:FUNCtion:BAND:SPAN:AUTO?

:CALCulate:MARKer<n>:MAXimum

Syntax
:CALCulate:MARKer<n>:MAXimum

Description
Performs one peak search based on the search mode set by the *:CALCulate:MARKer:PEAK:SEARch:MODE* command and marks it with the specified marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

Remarks
When no peak is found, a prompt message "No peak found" is displayed on the screen.

Example
The following command performs one peak search, and marks with Marker 2.
:CALCulate:MARKer2:MAXimum

:CALCulate:MARKer<n>:MAXimum:LEFT

Syntax
:CALCulate:MARKer<n>:MAXimum:LEFT

Description
Searches for and marks the nearest peak which is located at the left side of the current peak and meets the peak search condition.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

Remarks
When no peak is found, a prompt message "No peak found" is displayed on the screen.

Example
The following command performs one left peak search, and marks with Marker 2.
:CALCulate:MARKer2:MAXimum:LEFT

:CALCulate:MARKer<n>:MAXimum:NEXT

Syntax
:CALCulate:MARKer<n>:MAXimum:NEXT

Description
Searches for and marks the peak whose amplitude on the trace is next to that of the current peak and which meets the peak search condition.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

Remarks
When no peak is found, a prompt message "No peak found" is displayed on the screen.

Example
The following command performs one next peak search, and marks with Marker 2.
:CALCulate:MARKer2:MAXimum:NEXT

:CALCulate:MARKer<n>:MAXimum:RIGHt

Syntax
:CALCulate:MARKer<n>:MAXimum:RIGHt

Description
Searches for and marks the nearest peak which is located at the right side of the current peak and meets the peak search condition.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

Remarks
When no peak is found, a prompt message "No peak found" is displayed on the screen.

Example
The following command performs one right peak search, and marks with Marker 2.
:CALCulate:MARKer2:MAXimum:RIGHt

:CALCulate:MARKer<n>:MINimum

Syntax
:CALCulate:MARKer<n>:MINimum

Description
Searches for and marks the peak with the minimum amplitude on the trace.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

Remarks
When no peak is found, a prompt message "No peak found" is displayed on the screen.

Example
The following command performs one minimum search, and marks it with Marker 2.
:CALCulate:MARKer2:MINimum

:CALCulate:MARKer<n>[:SET]:CENTer

Syntax
:CALCulate:MARKer<n>[:SET]:CENTer

Description
Sets the frequency of the specified marker to be the center frequency of the analyzer.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

Remarks

If the marker mode of the specified marker is Position or Fixed, the center frequency will be set to the frequency of the marker.
If the specified marker mode is Delta, the center frequency will be set to the frequency of the Delta marker.
This function is invalid in zero span.
This command is invalid for the PvT measurement mode in RTSA.

Example
The following command sets the frequency of Marker 1 (Position) to the center frequency.
:CALCulate:MARKer1:SET:CENTer

:CALCulate:MARKer<n>[:SET]:DELTa:CENTer

Syntax
:CALCulate:MARKer<n>[:SET]:DELTa:CENTer

Description
Sets the frequency difference of the specified Delta marker to the center frequency of the analyzer.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

Remarks
It is only valid when the current marker mode is "Delta".
This function is invalid in zero span.
This command is invalid for the PvT measurement mode in RTSA.

Example
The following command sets the frequency difference of the Delta Marker 1 to the center frequency of the analyzer.
:CALCulate:MARKer1:SET:DELTa:CENTer

:CALCulate:MARKer<n>[:SET]:DELTa:SPAN

Syntax
:CALCulate:MARKer<n>[:SET]:DELTa:SPAN

Description
Sets the frequency difference of the specified Delta marker to the span of the analyzer.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

Remarks
It is only valid when the current marker mode is "Delta".
This function is invalid in zero span.
This command is invalid for the PvT measurement mode in RTSA.

Example
The following command sets the frequency difference of the Delta Marker 1 to the span of the analyzer.
:CALCulate:MARKer1:SET:DELTa:SPAN

:CALCulate:MARKer<n>[:SET]:RLEVel

Syntax
:CALCulate:MARKer<n>[:SET]:RLEVel

Description
Sets the amplitude of the specified marker to the reference level of the analyzer.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

Remarks
If the marker mode of the specified marker is Position or Fixed, the reference level will be set to the amplitude of the marker.
If the specified marker mode is Delta and the current marker is the reference marker, then the reference level is set to the amplitude of the reference marker; if the current marker is the Delta marker, then the reference level is set to the amplitude of the Delta marker.

Example
The following command sets the amplitude of Marker 2 (Position) to the reference level of the analyzer.
:CALCulate:MARKer2:SET:RLEVel

:CALCulate:MARKer<n>[:SET]:STARt

Syntax
:CALCulate:MARKer<n>[:SET]:STARt

Description
Sets the frequency of the specified marker to the start frequency of the analyzer.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

Remarks
If the marker mode of the specified marker is Position or Fixed, the start frequency will be set to the frequency of the marker.
If the specified marker mode is Delta, the start frequency will be set to the frequency of the Delta marker.
This function is invalid in zero span.
This command is invalid for the PvT measurement mode in RTSA.

Example
The following command sets the frequency of Marker 3 (Position) to the start frequency of the analyzer.
:CALCulate:MARKer3:SET:STARt

:CALCulate:MARKer<n>[:SET]:STEP

Syntax
:CALCulate:MARKer<n>[:SET]:STEP

Description
Sets the frequency of the specified marker to the center frequency step of the analyzer.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

Remarks
If the marker mode of the specified marker is Position or Fixed, the center frequency step will be set to the frequency of the marker.
If the specified marker mode is Delta, the center frequency step will be set to the frequency difference between the Delta marker and the reference marker.
This function is invalid in zero span.
This command is invalid for the PvT measurement mode in RTSA.

Example
The following command sets the frequency of Marker 4 (Position) to the center frequency step of the analyzer.
:CALCulate:MARKer4:SET:STEP

:CALCulate:MARKer<n>[:SET]:STOP

Syntax
:CALCulate:MARKer<n>[:SET]:STOP

Description
Sets the frequency of the specified marker to the stop frequency of the analyzer.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

Remarks
If the marker mode of the specified marker is Position or Fixed, the stop frequency will be set to the frequency of the marker.
If the specified marker mode is Delta, the stop frequency will be set to the frequency of the Delta marker.
This function is invalid in zero span.
This command is invalid for the PvT measurement mode in RTSA.

Example
The following command sets the frequency of Marker 2 (Position) to the stop frequency of the analyzer.
:CALCulate:MARKer2:SET:STOP

:CALCulate:MARKer<n>:TRACe

Syntax
:CALCulate:MARKer<n>:TRACe <integer>
:CALCulate:MARKer<n>:TRACe?

Description
Sets the marker trace for the specified marker.
Queries the marker trace for the specified marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

<integer>	Discrete	1|2|3|4|5|6|7	1

Remarks
<integer> indicates the marker trace, and its range is from Trace 1 to Trace 6. The selected trace must be enabled.
You can also run the *:CALCulate:MARKer<n>:TRACe:AUTO* command to set the marker trace of the specified marker to "Auto".
Only when the parameter <integer> is set to 7, can you set the marker trace in PvT view of RTSA mode. At this time, "PvT Trace1" is selected as the marker trace.

Return Format
The query returns any integer ranging from 1 to 6.
If the marker trace is set to "Auto", the query command returns a trace number on which the marker is placed.

Example
The following command sets the marker trace of Marker 1 to Trace 2.
:CALCulate:MARKer1:TRACe 2

The following query returns 2.
:CALCulate:MARKer1:TRACe?

:CALCulate:MARKer<n>:TRACe:AUTO

Syntax
:CALCulate:MARKer<n>:TRACe:AUTO <bool>
:CALCulate:MARKer<n>:TRACe:AUTO?

Description
Enables or disables the auto trace marking of the specified marker.
Queries the status of the auto trace marking of the specified marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
When you disable the auto marking of the trace, the currently enabled marker will stay on the corresponding trace.
This command is not supported in PvT view.

Return Format
The query returns 0 or 1.

Example
The following command sets the marker trace of Marker 1 to be Auto.
:CALCulate:MARKer1:TRACe:AUTO ON or :CALCulate:MARKer1:TRACe:AUTO 1

The following query returns 1.
:CALCulate:MARKer1:TRACe:AUTO?

:CALCulate:MARKer<n>:X

Syntax
:CALCulate:MARKer<n>:X <param>
:CALCulate:MARKer<n>:X?

Description
Sets the X-axis value of the specified marker. Its default unit is Hz.
Queries the X-axis value of the specified marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<param>	Real	Refer to "**Remarks**"	——

Remarks
If the readout mode is frequency, it cannot exceed the frequency range of the sweep and the units available are Hz (default), kHz, MHz, and GHz.
If the readout mode is time, it cannot exceed the time range of the sweep and the units available are s (default), us, ms, and ks.
<param> can be any value within the available range of the current X axis.
If the marker mode of the specified marker is Position or Fixed, this command sets the X value of the marker.
If the marker mode is set to Delta, this value sets the X-axis value of the Delta marker in relative to that of the reference marker.
The PvT only supports setting the marker on Trace 1.

Return Format
The query returns the X-axis value of the marker in scientific notation.

Example
The following command sets the X-axis value of Marker 1 to 150 MHz (the readout mode is frequency).
:CALCulate:MARKer1:X 150000000

The following query returns 1.500000000e+08.
:CALCulate:MARKer1:X?

:CALCulate:MARKer<n>:X:READout

Syntax
:CALCulate:MARKer<n>:X:READout <type>
:CALCulate:MARKer<n>:X:READout?

Description
Sets the readout mode of the X axis of the specified marker..
Queries the readout mode of the X axis of the specified marker..

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<type>	Discrete	FREQuency|TIME|ITIMe|PERiod	Refer to "**Remarks**"

Remarks
FREQuency: indicates frequency. It is the default readout mode in non-zero span mode.
TIME: indicates time. It is the default readout mode in zero span mode.
ITIMe: indicates the reciprocal of time. It is available only in zero span mode and a Delta marker is selected.
PERiod: indicates period. It is not available in zero span mode.
This command is not supported in PvT view.

Return Format

The query returns FREQ, TIME, ITIM, or PER.

Example
The following command sets the readout mode of the X axis of Marker 1 to "Time".
:CALCulate:MARKer1:X:READout TIME

The following query returns TIME.
:CALCulate:MARKer1:X:READout?

:CALCulate:MARKer<n>:X:READout:AUTO

Syntax
:CALCulate:MARKer<n>:X:READout:AUTO <bool>
:CALCulate:MARKer<n>:X:READout:AUTO?

Description
Enables or disables the auto readout mode of the specified marker.
Queries the auto readout mode of the specified marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
When the auto readout mode is enabled, if the marker trace changes, the readout mode will be re-determined based on the destination trace.
This command is not supported in PvT view.

Return Format
The query returns 0 or 1.

Example
The following command enables the auto readout mode of Marker 1 in the X-axis.
:CALCulate:MARKer1:X:READout:AUTO ON or :CALCulate:MARKer1:X:READout:AUTO 1

The following query returns 1.
:CALCulate:MARKer1:X:READout:AUTO?

:CALCulate:MARKer<n>:X:POSition

Syntax
:CALCulate:MARKer<n>:X:POSition <index>
:CALCulate:MARKer<n>:X:POSition?

Description
Sets the X-axis position of the trace at the specified marker.
Queries the X-axis position of the trace at the specified marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<index>	Integer	——	——

Remarks:
N/A.

Return Format
The query returns the X-axis position of the trace at the specified marker in integer.

Example
The following command sets the X-axis position of the trace at Marker 1 to 100.
:CALCulate:MARKer1:X 100

The following query returns 100.
:CALCulate:MARKer1:X?

:CALCulate:MARKer<n>:Y

Syntax
:CALCulate:MARKer<n>:Y <amp>
:CALCulate:MARKer<n>:Y?

Description
Sets the Y-axis value of the specified marker.
Queries the Y-axis value of the specified marker. Its default unit is dBm.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<amp>	Real	-170 dBm to +30 dBm	——

Remarks
This setting command is only available when there is only one marker and its marker mode is Fixed.
If the marker mode of the specified marker is Delta, the query command queries the Y-axis difference between the reference marker and the Delta marker.

Return Format
The query returns the Y-axis value in scientific notation.

Example
The following command sets Y-axis value at Fixed Marker 1 to -59.6 dBm.
:CALCulate:MARKer1:Y -59.6

The following query returns -5.960000000e+01.
:CALCulate:MARKer1:Y?

:CALCulate:MARKer<n>:Z?

Syntax
:CALCulate:MARKer<n>:Z?

Description
Queries the Z-axis value of the specified marker.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1

Remarks
This command is valid when the Waterfall winow is enabled and the marker trace is not PvT trace.

Example
The following command returns the Z-axis value of Marker 1.
:CALCulate:MARKer1:Z?

:CALCulate:MARKer<n>:Z:POSition

Syntax
:CALCulate:MARKer<n>:Z:POSition <integer>
:CALCulate:MARKer<n>:Z:POSition?

Description
Sets the trace number of the specified marker in the Spectrogram view.
Queries the trace number of the specified marker in the Spectrogram view.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8	1
<integer>	Integer	1 to 8192	1

Remarks
This command is only valid when you select the Waterfall window in the Spectrogram view, Density view, or PvT Spectrogram view in RTSA mode.

Return Format
The query returns the trace number in integer.

Example
The following command sets the trace number at Marker 1 to 100.
:CALCulate:MARKer1:Z:POSition 100

The following query returns 100.
:CALCulate:MARKer1:Z:POSition?

:CALCulate:MATH

Syntax
:CALCulate:MATH
<trace_destination>,<function_math>,<trace1_operation>,<trace2_operation>,<rel_offset>,<rel_reference>
:CALCulate:MATH? TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6

Description
Sets mathematical operations between traces and, in some cases, user-defined offsets.
Queries the math operation function of the specified trace.

Parameter

Name	Type	Range	Default
<trace_destination>	Discrete	TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6	——
<function_math>	Discrete	PDIFference|PSUM|LOFFset|LMOFFset|LDIFference|OFF	OFF
<trace1_operation>	Discrete	TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6	TRACE5
<trace2_operation>	Discrete	TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6	TRACE6
<rel_offset>	Real	-100 dB to 100 dB	0 dB
<rel_reference>	Real	-170 dBm to 30 dBm	0 dBm

Remarks
PDIFference: indicates the power difference between Operand1 (Op1) and Operand 2 (Op2).
PSUM: indicates the power sum between Operand 1 (Op1) and Operand 2 (Op2).

LOFFset: indicates the sum between Operand 1 (Op1) and Offset (<rel_offset>).
LMOFFset: indicates the difference between Operand 1 (Op1) and Offset (<rel_offset>).
LDIFference: Operand 1 minus Operand 2, and then plus the reference value (<rel_reference>).
OFF: disables the math operation function.
<rel_offset> indicates the log offset, and its unit is dB; <rel_reference> indicates the log reference, and its unit is dBm.
For certain operation, if some parameters are irrelevant in the operation, you can replace these parameters with ",," in the command. The returned parameters are separated by commas. The returned results of the irrelevant parameters are undefined. The empty fields are replaced by ",,".

Return Format
The query returns PDIF, PSUM, LOFF, LMOFF, LDIF, OFF.

Example
The following command sets Trace 1 to be the power difference math operation, and sets Operand Trace 1 to be Trace 4 and Operand Trace 2 to be Trace 5.
:CALCulate:MATH TRACE1,PDIFference,TRACE4,TRACE5,,

The following query returns PDIF.
:CALCulate:MATH? TRACE1

:CALCulate:NTData[:STATe]

Syntax
:CALCulate:NTData[:STATe] <bool>
:CALCulate:NTData[:STATe]?

Description
Enables or disables the normalize function.
Queries the status of normalization.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only available in GPSA mode.
This command is only valid when the tracking generator function is enabled.

Return Format
The query returns 0 or 1.

Example
The following command enables normalization.
:CALCulate:NTData:STATe ON or :CALCulate:NTData:STATe 1

The following query returns 1.
:CALCulate:NTData:STATe?

:CALCulate:OBWidth:LIMit:FBLimit

Syntax
:CALCulate:OBWidth:LIMit:FBLimit <freq>
:CALCulate:OBWidth:LIMit:FBLimit?

Description
Sets the limit value in OBW measurement.

Queries the limit value in OBW measurement.

Parameter

Name	Type	Range	Default
<freq>	Real	1 kHz to Fmax	5 MHz

Remarks
This command is only valid for the OBW measurement in GPSA mode.

Return Format
The query returns the limit value in integer.

Example
The following command sets the limit value to 1 kHz.
:CALCulate:OBWidth:LIMit:FBLimit 1

The following query returns 1000.000000.
:CALCulate:OBWidth:LIMit:FBLimit?

:CALCulate:OBWidth:LIMit[:TEST]

Syntax
:CALCulate:OBWidth:LIMit[:TEST] <bool>
:CALCulate:OBWidth:LIMit[:TEST]?

Description
Sets the limit state in OBW measurement.
Queries the limit state in OBW measurement.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only valid for the OBW measurement in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the limit function.
:CALCulate:OBWidth:LIMit 1 or :CALCulate:OBWidth:LIMit ON

The following query returns 1000.000000.
:CALCulate:OBWidth:LIMit?

:CALCulate:LIMit<n>:FAIL?

Syntax
:CALCulate:LIMit<n>:FAIL?

Description
Queries the result of the limit line test.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1

:CALCulate:TRACe<n>:FAIL?

Syntax
:CALCulate:TRACe<n>:FAIL?

Description
Queries the result of the trace line test.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1

:CALibration Commands

:CALibration[:ALL]

Syntax
:CALibration[:ALL]

Description
Executes self-calibration immediately.

Example
The following command executes the self-calibration immediately.
:CALibration:ALL

:CALibration:AUTO

Syntax
:CALibration:AUTO <bool>
:CALibration:AUTO?

Description
Enables or disables auto calibration.
Queries the setting status of auto calibration.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 0 or 1.

Example
The following command enables auto calibration.
:CALibration:AUTO ON or :CALibration:AUTO 1

The following query returns 1.
:CALibration:AUTO?

:CONFigure Commands

:CONFigure?

Syntax
:CONFigure?

Description
Queries the current measurement function.

Return Format
The query returns SA, RTSA, EMI, VSA, AM, FM, PM, GPSA_TPOW, GPSA_ACP, GPSA_CHP, GPSA_OBW, GPSA_CNR, GPSA_HARM, GPSA_TOI, GPSA_TG, or RTSA_UBW_SCAN.

:CONFigure:CATalog?

Syntax
:CONFigure:CATalog?

Description
Queries the currently supported measurement application.

:COUPle Commands

:COUPle

Syntax
:COUPle <enum>

Description
Sets auto coupling function.

Parameter

Name	Type	Range	Default
<enum>	Discrete	ALL|NONE	-

Remarks
ALL: enables auto coupling.
NONE: disables auto coupling. Couple manually.

Return Format
N/A

Example
The following command disables auto coupling function.
:COUPle NONE

:DISPlay Commands

:DISPlay:PVTime:WINDow:TRACe:X[:SCALe]:COUPle

Syntax
:DISPlay:PVTime:WINDow:TRACe:X[:SCALe]:COUPle <bool>
:DISPlay:PVTime:WINDow:TRACe:X[:SCALe]:COUPle?

Description
Enables or disables the auto scale function for the horizontal axis in the PvT view.
Queries the status of the auto scale function for the horizontal axis in the PvT view.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
This command is only valid when the PvT view in RTSA mode is selected.

Return Format
The query returns 1 or 0.

Example
The following command enables the auto scale function for the horizontal axis in the PvT view.
:DISPlay:PVTime:WINDow:TRACe:X:SCALe:COUPle ON
or :DISPlay:PVTime:WINDow:TRACe:X:SCALe:COUPle 1

The following query returns 1.
:DISPlay:PVTime:WINDow:TRACe:X:SCALe:COUPle?

:DISPlay:PVTime:WINDow:TRACe:X[:SCALe]:PDIVision

Syntax
:DISPlay:PVTime:WINDow:TRACe:X[:SCALe]:PDIVision <time>
:DISPlay:PVTime:WINDow:TRACe:X[:SCALe]:PDIVision?

Description
Sets the unit per division in the horizontal axis of the PvT view.
Queries the unit per division in the horizontal axis of the PvT view.

Parameter

Name	Type	Range	Default
<time>	Real	20 us to 4 s	Acquisition Time/10

Remarks
This command is only valid when the PvT view in RTSA mode is selected.

Return Format
The query returns the X-axis scale value in scientific notation.

Example
The following command sets the horizontal scale in the PvT view to 40 us.
:DISPlay:PVTime:WINDow:TRACe:X:SCALe:PDIVision 0.00004

The following query returns 4.000000000e-07.

:DISPlay:PVTime:WINDow:TRACe:X:SCALe:PDIVision?

:DISPlay:PVTime:WINDow:TRACe:X[:SCALe]:RLEVel

Syntax
:DISPlay:PVTime:WINDow:TRACe:X[:SCALe]:RLEVel <real>
:DISPlay:PVTime:WINDow:TRACe:X[:SCALe]:RLEVel?

Description
Sets the reference time for the horizontal axis in the PvT view.
Queries the reference time for the horizontal axis in the PvT view.

Parameter

Name	Type	Range	Default
<real>	Real	-1 s to 40 s	0 μs

Remarks
This command is only valid when the PvT view in RTSA mode is selected.

Return Format
The query returns the reference time for the horizontal axis in scientific notation.

Example
The following command sets the reference time for the horizontal axis in the PvT view to 2 s.
:DISPlay:PVTime:WINDow:TRACe:X:SCALe:RLEVel 2

The following query returns 2.000000000e+00.
:DISPlay:PVTime:WINDow:TRACe:X:SCALe:RLEVel?

:DISPlay:PVTime:WINDow:TRACe:X[:SCALe]:RPOSition

Syntax
:DISPlay:PVTime:WINDow:TRACe:X[:SCALe]:RPOSition <position>
:DISPlay:PVTime:WINDow:TRACe:X[:SCALe]:RPOSition?

Description
Sets the position of the reference time in the horizontal axis of the PvT view.
Queries the position of the reference time in the horizontal axis of the PvT view.

Parameter

Name	Type	Range	Default
<position>	Discrete	LEFT|CENTer|RIGHt	LEFT

Remarks
This command is only valid when the PvT view in RTSA mode is selected.

Return Format
The query returns LEFT, CENT, or RIGH.

Example
The following command sets the position of the reference time in the horizontal axis of the PvT view to "Left".
:DISPlay:PVTime:WINDow:TRACe:X:SCALe:RPOSition LEFT
The following query returns LEFT.
:DISPlay:PVTime:WINDow:TRACe:X:SCALe:RPOSition?

:DISPlay:VIEW[:SELect]

Syntax
:DISPlay:VIEW[:SELect] <view>
:DISPlay:VIEW[:SELect]?

Description
Sets the current display view.
Queries the current display view.

Parameter

Name	Type	Range	Default
<view>	Discrete	NORMal|SPECtrogram|DENSity|DSGRam|PVTime|PSPectrum|PSGRam|CNR|ACP|TOI|CHP|TP|OBW|HD	NORMal

Remarks
- The valid parameters in RTSA mode are NORMal, SPECtrogram, DENSity, DSGRam, PVTime, PSPectrum, and PSGRam.
  NORMal: indicates normal view.
  SPECtrogram: indicates Spectrogram view.
  DENSity: indicates Density view.
  DSGRam: indicates Density Spectrogram view.
  PVTime: indicates Power vs Time view.
  PSPectrum: indicates PvT Spectrum view.
  PSGRam: indicates PvT Spectrogram view.
- The valid parameters in GPSA mode are
  CNR: indicates C/N ratio.
  ACP: indicates adjacent power.
  TOI: indicates third-order Intercept.
  CHP: indicates channel power.
  TP: indicates time-domain power.
  OBW: indicates occupied bandwidth.
  HD: indicates harmonic distortion.

Return Format
The query returns NORM, SPEC, DENS, DSGR, PVT, PVTS, PSGR, PSP, CNR, ACP, TOI, CHP, TP, OBW, or HD.

Example
The following commands sets the current view to Spectrogram.
:DISPlay:VIEW:SELect SPECtrogram

The following query returns SPEC.
:DISPlay:VIEW:SELect?

:DISPlay:VIEW:DENSity:PERSistence

Syntax
:DISPlay:VIEW:DENSity:PERSistence <time>
:DISPlay:VIEW:DENSity:PERSistence?

Description
Sets the persistence in the Density view.
Queries the persistence in the Density view.

Parameter

Name	Type	Range	Default
<time>	Real	0 s to 10 s	300 ms

Remarks
This command is only available for Density and Density Spectrogram views in RTSA mode.

Return Format
The query returns the persistence time in scientific notation.

Example
The following command sets the persistence in the Density view to 5 s.
:DISPlay:VIEW:DENSity:PERSistence 5

The following query returns 5.000000e+00.
:DISPlay:VIEW:DENSity:PERSistence?

:DISPlay:VIEW:DENSity:PERSistence:INFinite

Syntax
:DISPlay:VIEW:DENSity:PERSistence:INFinite <bool>
:DISPlay:VIEW:DENSity:PERSistence:INFinite?

Description
Enables or disables the infinite mode of the persistence time.
Queries the on/off status of the infinite mode of the persistence time.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only available for Density and Density Spectrogram views in RTSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the infinite mode of the persistence time.
:DISPlay:VIEW:DENSity:PERSistence:INFinite ON or :DISPlay:VIEW:DENSity:PERSistence:INFinite 1

The following query returns 1.
:DISPlay:VIEW:DENSity:PERSistence:INFinite?

:DISPlay:VIEW:DENSity:HDHue

Syntax
:DISPlay:VIEW:DENSity:HDHue <real>
:DISPlay:VIEW:DENSity:HDHue?

Description
Sets the highest density hue.
Queries the highest density hue.

Parameter

Name	Type	Range	Default
<real>	Real	0.1 to 100	100

Remarks
This command is only available for Density and Density Spectrogram views in RTSA mode.

Return Format
The query returns the highest density hue.

Example
The following command sets the highest density hue in the Density view to 60.
:DISPlay:VIEW:DENSity:HDHue 60

The following query returns 6.000000e+01.
:DISPlay:VIEW:DENSity:HDHue?

:DISPlay:VIEW:DENSity:LDHue

Syntax
:DISPlay:VIEW:DENSity:LDHue <real>
:DISPlay:VIEW:DENSity:LDHue?

Description
Sets the lowest density hue.
Queries the lowest density hue.

Parameter

Name	Type	Range	Default
<real>	Real	0 to 99.9	0

Remarks
This command is only available for Density and Density Spectrogram views in RTSA mode.

Return Format
The query returns the lowest density hue.

Example
The following command sets the lowest density hue in the Density view to 30.
:DISPlay:VIEW:DENSity:LDHue 30

The following query returns 3.000000e+01.
:DISPlay:VIEW:DENSity:LDHue?

:DISPlay:VIEW:DENSity:CNONlinear

Syntax
:DISPlay:VIEW:DENSity:CNONlinear <real>
:DISPlay:VIEW:DENSity:CNONlinear?

Description
Sets the curve nonlinearity.
Queries the curve nonlinearity.

Parameter

Name	Type	Range	Default
<real>	Real	-100 to 100	75

Remarks
This command is only available for Density and Density Spectrogram views in RTSA mode.
Within the range between the highest density hue and lowest density hue, setting the curve nonlinearity can change the gradient among different density hues, making the displayed results move towards either the higher or lower end of the gradient. Increasing the curve nonlinearity value will compress the colors towards the higher end of the color bar, and decreasing the nonlinearity value will compress the colors towards the lower end of the color bar.

Return Format
The query returns the curve nonlinearity in scientific notation.

Example
The following command sets the curve nonlinearity to 50.
:DISPlay:VIEW:DENSity:CNONlinear 50

The following query returns 5.000000e+01.
:DISPlay:VIEW:DENSity:CNONlinear?

:DISPlay:VIEW:DENSity:CPALettes

Syntax
:DISPlay:VIEW:DENSity:CPALettes <capl>
:DISPlay:VIEW:DENSity:CPALettes?

Description
Sets the color palette of Density view.
Queries the color palette of Density view.

Parameter

Name	Type	Range	Default
<capl>	Discrete	COOL|WARM|RADar|FIRE|FROSt	WARM

Remarks
This command is only available for Density and Density Spectrogram views in RTSA mode.

Return Format
The query returns COOL, WARM, RAD, FIRE, or FROS.

Example
The following command sets the color palette in the Density view to Cool.
:DISPlay:VIEW:DENSity:CPALettes COOL

The following query returns COOL.
:DISPlay:VIEW:DENSity:CPALettes?

:DISPlay:VIEW:DENSity:AADJust

Syntax
:DISPlay:VIEW:DENSity:AADJust

Description
Sets the highest density hue to the highest density value found in the current bitmap; sets the lowest density

hue to the lowest non-zero density value found in the current bitmap.

Remarks
This command is only available for Density and Density Spectrogram views in RTSA mode.

:DISPlay:VIEW:DENSity:TRUNcation

Syntax
:DISPlay:VIEW:DENSity:TRUNcation <bool>
:DISPlay:VIEW:DENSity:TRUNcation?

Description
Sets the truncation state in the Density view.
Queries the truncation state in the Density view.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only available for Density and Density Spectrogram views in RTSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the infinite mode of the persistence time.
:DISPlay:VIEW:DENSity:TRUNcation ON or :DISPlay:VIEW:DENSity:TRUNcation 1

The following query returns 1.
:DISPlay:VIEW:DENSity:TRUNcation?

:DISPlay:VIEW:SPECtrogram:TRACe:SELection

Syntax
:DISPlay:VIEW:SPECtrogram:TRACe:SELection <type>
:DISPlay:VIEW:SPECtrogram:TRACe:SELection?

Description
Sets the trace display type in the Spectrogram view.
Queries the trace display type in the Spectrogram view.

Parameter

Name	Type	Range	Default
<type>	Discrete	TIME|TNUMber	OFF|0

Remarks
TIME: indicates the trace time.
TNUMber: indicates the trace number.
This command is only valid in RTSA mode.

Return Format
The query returns TIME or TNUM.

Example
The following command sets the trace display type in the Spectrogram view to TIME.

:DISPlay:VIEW:SPECtrogram:TRACe:SELection TIME

The following query returns TIME.
:DISPlay:VIEW:SPECtrogram:TRACe:SELection?

:DISPlay:WINDow<n>:TRACe:X[:SCALe]:SPACing

Syntax
:DISPlay:WINDow<n>:TRACe:X[:SCALe]:SPACing <enum>
:DISPlay:WINDow<n>:TRACe:X[:SCALe]:SPACing?

Description
Sets the X-axis scale type.
Queries the X-axis scale type.

Parameter

Name	Type	Range	Default
<n>	Discrete	1	--
<enum>	Discrete	LINear|LOGarithmic	LOGarithmic

Remarks
1: Normal measurement and advanced measurement in GPSA mode; normal view in RTSA mode
LINear: indicates linear.
LOGarithmic: indicates log.

Return Format
The query returns LIN or LOG.

Example
The following command sets the X-axis scale type to LOG.
:DISPlay:WINDow1:TRACe:X:SCALe:SPACing LOGarithmic

The following query returns LOG.
:DISPlay:WINDow1:TRACe:X:SCALe:SPACing?

:DISPlay:WINDow<n>:TRACe:X:FLINe <flnum>

Syntax
:DISPlay:WINDow<n>:TRACe:X:FLINe<flnum> <ampl>
:DISPlay:WINDow<n>:TRACe:X:FLINe<flnum>?

Description
Sets the position of the frequency line.
Queries the position of the frequency line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|7	——
<flnum>	Discrete	1|2|3|4	1
<ampl>	Real	0 Hz to F_{max}	1 GHz

Remarks
Fmax is determined by the instrument model. Refer to the product model list in the Overview chapter of the document.
The definition for the parameter <n> is as follows:
1: Normal measurement and advanced measurement in GPSA mode; normal view in RTSA mode

7: Density view in RTSA mode

Return Format
The query returns the position of frequency line in integer.

Example
The following command sets the position of frequency line 1 to 4.21 GHz.
:DISPlay:WINDow1:TRACe:X:FLINe1 421000000

The following query returns 421000000.
:DISPlay:WINDow1:TRACe:X:FLINe1?

:DISPlay:WINDow<n>:TRACe:X:FLINe <flnum>:STATe

Syntax
:DISPlay:WINDow<n>:TRACe:X:FLINe <flnum>:STATe <bool>
:DISPlay:WINDow<n>:TRACe:X:FLINe <flnum>:STATe?

Description
Sets the state of the frequency line.
Queries the state of the frequency line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|7	——
<flnum>	Discrete	1|2|3|4	1
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
The definition for the parameter <n> is as follows:
1: Normal measurement and advanced measurement in GPSA mode; normal view in RTSA mode
7: Density view in RTSA mode

Return Format
The query returns 0 or 1.

Example
The following command enables Frequency Line 1.
:DISPlay:WINDow1:TRACe:X:FLINe1:STATe 1 or :DISPlay:WINDow1:TRACe:X:FLINe1:STATe ON

The following query returns 1.
:DISPlay:WINDow1:TRACe:X:FLINe1:STATe?

:DISPlay:WINDow<n>:TRACe:X:TLINe<tlnum>

Syntax
:DISPlay:WINDow<n>:TRACe:X:TLINe<tlnum> <ampl>
:DISPlay:WINDow<n>:TRACe:X:TLINe<tlnum>?

Description
Sets the position of the time line.
Queries the position of the time line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|8	——
<tlnum>	Discrete	1|2|3|4	1
<ampl>	Real	0 s to 1 Ps	1 ms

Remarks
The definition for the parameter <n> is as follows:
1: Normal measurement in GPSA mode (zero span)
8: PvT trace view in RTSA mode

Return Format
The query returns the position of time line in integer.

Example
The following command sets the position of Time Line 1 to 0 s.
:DISPlay:WINDow1:TRACe:X:TLINe1 0

The following query returns 0.
:DISPlay:WINDow1:TRACe:X:TLINe1?

:DISPlay:WINDow<n>:TRACe:X:TLINe<tlnum>:STATe

Syntax
:DISPlay:WINDow<n>:TRACe:X:TLINe<tlnum>:STATe <bool>
:DISPlay:WINDow<n>:TRACe:X:TLINe<tlnum>:STATe?

Description
Sets the state of the time line.
Queries the state of the time line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|8	——
<tlnum>	Discrete	1|2|3|4	1
<ampl>	Real	0 s to 1 Ps	1 ms

Remarks
The definition for the parameter <n> is as follows:
1: Normal measurement in GPSA mode (zero span)
8: PvT trace view in RTSA mode

Return Format
The query returns 0 or 1.

Example
The following command enables Time Line 1.
:DISPlay:WINDow1:TRACe:X:TLINe1:STATe 1 or :DISPlay:WINDow1:TRACe:X:TLINe1:STATe ON

The following query returns 1.
:DISPlay:WINDow1:TRACe:X:TLINe1:STATe?

:DISPlay:WINDow<n>:TRACe:Y:DLINe<dlnum>

Syntax
:DISPlay:WINDow<n>:TRACe:Y:DLINe<dlnum> <ampl>
:DISPlay:WINDow<n>:TRACe:Y:DLINe<dlnum>?

Description
Sets the position of the display line. Its default unit is dBm.
Queries the position of the display line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|7|8	——
<dlnum>	Discrete	1|2|3|4	1
<ampl>	Real	Current amplitude range	-25 dBm

Remarks
By default, the display line is disabled. When the display line is enabled for the first time, its position is -25 dBm.
1: Normal measurement and advanced measurement in GPSA mode; normal view in RTSA mode
7: Density view in RTSA mode
8: PvT trace view in RTSA mode

Return Format
The query returns the position of the display line in scientific notation.

Example
The following command sets the position of the display line1 to -10 dBm.
:DISPlay:WINDow1:TRACe:Y:DLINe1 -10

The following command returns -1.000000e+01.
:DISPlay:WINDow1:TRACe:Y:DLINe1?

:DISPlay:WINDow<n>:TRACe:Y:DLINe<dlnum>:STATe

Syntax
:DISPlay:WINDow<n>:TRACe:Y:DLINe<dlnum>:STATe <bool>
:DISPlay:WINDow<n>:TRACe:Y:DLINe<dlnum>:STATe?

Description
Enables or disables the display line.
Queries the on/off status of the display line.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|7|8	——
<dlnum>	Discrete	1|2|3|4	1
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
1: Normal measurement and advanced measurement in GPSA mode; normal view in RTSA mode
7: Density view in RTSA mode
8: PvT trace view in RTSA mode

Return Format
The query returns 0 or 1.

Example
The following command enables the display line1.
:DISPlay:WINdow1:TRACe:Y:DLINe1:STATe ON or :DISPlay:WINdow1:TRACe:Y:DLINe1:STATe 1

The following query returns 1.
:DISPlay:WINdow1:TRACe:Y:DLINe1:STATe?

:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:NRLevel

Syntax
:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:NRLevel <rel_ampl>
:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:NRLevel?

Description
Sets the reference level of normalization.
Queries the reference level of normalization.

Parameter

Name	Type	Range	Default
<n>	Discrete	1	--
<rel_ampl>	Real	-200 dB to 200 dB	10 dB

Remarks
This command is only available in GPSA mode.
This command is only valid when the tracking generator function is enabled.

Return Format
The query returns the reference level of normalization in scientific notation.

Example
The following command sets the reference level of normalization to -20 dB.
:DISPlay:WINDow1:TRACe:Y:SCALe:NRLevel -20

The following query returns -2.000000e+01.
:DISPlay:WINDow1:TRACe:Y:SCALe:NRLevel?

:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:NRPosition

Syntax
:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:NRPosition <integer>
:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:NRPosition?

Description
Sets the reference position of normalization.
Queries the reference position of normalization.

Parameter

Name	Type	Range	Default
<n>	Discrete	1	--
<integer>	Integer	0% to 100%	100%

Remarks
This command is only available in GPSA mode.
This command is only valid when the tracking generator function is enabled.

Return Format
The query returns the reference position of normalization in integer.

Example
The following command sets the reference position of normalization to 50%.
:DISPlay:WINDow1:TRACe:Y:SCALe:NRPosition 50

The following query returns 50.
:DISPlay:WINDow1:TRACe:Y:SCALe:NRPosition?

:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:PDIVision

Syntax
:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:PDIVision <rel_ampl>
:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:PDIVision?

Description
Sets the Y-axis scale type.
Queries the Y-axis scale type.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|4|7|8	——
<rel_ampl>	Real	0.1 dB to 20 dB	10 dB

Remarks
The definition for the parameter <n> is as follows:
1: Normal measurement and advanced measurement in GPSA mode; normal view in RTSA mode
4: waterfall window in RTSA mode
7: Density view in RTSA mode
8: PvT trace view in RTSA mode

Return Format
The query returns the Y-axis scale value in scientific notation.

Example
The following command sets the Y-axis scale value to 15 dB.
:DISPlay:WINDow1:TRACe:Y:SCALe:PDIVision 15

The following query returns 1.500000e+01.
:DISPlay:WINDow1:TRACe:Y:SCALe:PDIVision?

:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:RLEVel

Syntax
:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:RLEVel <ampl>
:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:RLEVel?

Description
Sets the reference level.
Queries the reference level.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|4|7|8	——
<ampl>	Real	-170 dBm to 30 dBm	0 dBm

Remarks
The definition for the parameter <n> is as follows:
1: Normal measurement and advanced measurement in GPSA mode; normal view in RTSA mode
4: waterfall window in RTSA mode
7: Density view in RTSA mode
8: PvT trace view in RTSA mode

Return Format
The query returns the reference level in scientific notation.

Example
The following command sets the reference level to -10 dB.
:DISPlay:WINDow1:TRACe:Y:SCALe:RLEVel -10

The following query returns -1.000000e+01.
:DISPlay:WINDow1:TRACe:Y:SCALe:RLEVel?

:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:RLEVel:OFFSet

Syntax
:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:RLEVel:OFFSet <rel_ampl>
:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:RLEVel:OFFSet?

Description
Sets the reference level offset.
Queries the reference level offset.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|4|7|8	——
<rel_ampl>	Real	-300 dB to 300 dB	0 dB

Remarks
The offset of the reference level only modifies the reference level and amplitude readout of the marker, but does not change the position of the curve.
The definition for the parameter <n> is as follows:
1: Normal measurement and advanced measurement in GPSA mode; normal view in RTSA mode
4: waterfall window in RTSA mode
7: Density view in RTSA mode
8: PvT trace view in RTSA mode

Return Format
The query returns the offset of the reference level in scientific notation.

Example
The following command sets the reference level offset to 10 dB.
:DISPlay:WINDow1:TRACe:Y:SCALe:RLEVel:OFFSet 10

The following query returns 1.000000e+01.
:DISPlay:WINDow1:TRACe:Y:SCALe:RLEVel:OFFSet?

:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:SPACing

Syntax
:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:SPACing <enum>
:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:SPACing?

Description
Sets the Y-axis scale type.
Queries the Y-axis scale type.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|4|7|8	——
<enum>	Discrete	LINear|LOGarithmic	LOGarithmic

Remarks
LINear: indicates linear.
LOGarithmic: indicates log.
The definition for the parameter <n> is as follows:
1: Normal measurement and advanced measurement in GPSA mode

Return Format
The query returns LIN or LOG.

Example
The following command sets the Y-axis scale type to LOGarithmic.
:DISPlay:WINDow1:TRACe:Y:SCALe:SPACing LOGarithmic

The following query returns LOG.
:DISPlay:WINDow1:TRACe:Y:SCALe:SPACing?

:DISPlay:WINDow4:TRACe:Position

Syntax
:DISPlay:WINDow4:TRACe:Position <position>
:DISPlay:WINDow4:TRACe:Position?

Description
Sets the display trace in the Waterfall view.
Queries the display trace in the Waterfall view.

Parameter

Name	Type	Range	Default
<position>	Integer	1 to 8,192	1

Remarks
This command is only available for RTSA mode.

Return Format
The query returns the display trace in integer.

Example
The following command sets the display trace to 5.
:DISPlay:WINDow4:TRACe:Position 5

The following query returns 5.
:DISPlay:WINDow4:TRACe:Position?

:DISPlay:WINDow4:TRACe:TIME

Syntax
:DISPlay:WINDow4:TRACe:TIME <time>
:DISPlay:WINDow4:TRACe:TIME?

Description
Sets the display trace in the Waterfall view when the display mode is Time. It is expressed in time.
Queries the display trace expressed in time in the Waterfall view when the display mode is Time.

Parameter

Name	Type	Range	Default
<time>	Real	-300 dB to 300 dB	0 dB

Remarks
This command is only available for RTSA mode.

Return Format
The query returns the display trace expressed in time in the Waterfall view in scientific notation.

Example
The following command sets the display trace to 10 s.
:DISPlay:WINDow4:TRACe:TIME 10

The following query returns 1.000000e+01.
:DISPlay:WINDow4:TRACe:TIME?

:DISPlay:WINDow4:TRACe:COUPle

Syntax
:DISPlay:WINDow4:TRACe:COUPle <bool>
:DISPlay:WINDow4:TRACe:COUPle?

Description
Sets the trace coupling relationship between marker Z and trace position.
Queries the trace coupling relationship between marker Z and trace position.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 0 or 1.

Example
The following command enables trace coupling.
:DISPlay:WINdow4:TRACe:COUPle ON or :DISPlay:WINdow:TRACe:COUPle 1

The following query returns 1.
:DISPlay:WINdow4:TRACe:COUPle?

:DISPlay:WINDow4:HUE

Syntax
:DISPlay:WINDow4:HUE <real>
:DISPlay:WINDow4:HUE?

Description
Sets the reference hue in the Waterfall window.
Queries the reference hue in the Waterfall window.

Parameter

Name	Type	Range	Default
<real>	Real	0 to 359.9	0

Remarks
This command is only available for RTSA mode.

Return Format
The query returns the reference hue in the Waterfall window in scientific notation.

Example
The following command sets the reference hue in the Waterfall window to 10.
:DISPlay:WINDow4:HUE 10

The following query returns 1.000000e+01.
:DISPlay:WINDow4:HUE?

:DISPlay:WINDow4:REFerence

Syntax
:DISPlay:WINDow4:REFerence<integer>
:DISPlay:WINDow4:REFerence?

Description
Sets the reference hue position in the Waterfall window.
Queries the reference hue position in the Waterfall window.

Parameter

Name	Type	Range	Default
<integer>	Integer	Max (10%, bottom hue value + 10%) to 100.0	100

Remarks
This command is only available for RTSA mode.

Return Format
The query returns the reference hue position in the Waterfall window in integer.

Example
The following command sets the reference hue position in the Waterfall window to 10.
:DISPlay:WINDow4:REFerence10

The following query returns 10.
:DISPlay:WINDow4:REFerence?

:DISPlay:WINDow4:BOTTom

Syntax
:DISPlay:WINDow4:BOTTom <integer>
:DISPlay:WINDow4:BOTTom?

Description
Sets the bottom hue position in the Waterfall window.
Queries the bottom hue position in the Waterfall window.

Parameter

Name	Type	Range	Default
<integer>	Real	0 to Min (90%, reference position value - 10%)	0

Remarks
This command is only available for RTSA mode.

Return Format
The query returns the bottom hue position in the Waterfall window in integer.

Example
The following command sets the bottom hue position in the Waterfall window to 10.
:DISPlay:WINDow4: BOTTom 10

The following query returns 10.
:DISPlay:WINDow4: BOTTom?

:DISPlay:WINDow4:AADJust

Syntax
:DISPlay:WINDow4:AADJust

Description
Auto adjusts the hue position in the Waterfall window.

:FETCh Commands

Note:
The :FETCh commands are only applicable to the RSA800 series that has been installed with advanced measurement option.

:FETCh:CNRatio?

Syntax
:FETCh:CNRatio?

Description
Queries the results of C/N ratio measurement.

Remarks
This command is only valid when the C/N ratio measurement is enabled in GPSA mode.
The power unit of the returned value is consistent with the current Y-axis unit.

Return Format
The query returns the carrier power, noise power, and C/N ratio (in dB) in scientific notation. They are separated by commas.

Example
The following query returns -6.048788000e+01,-6.186192000e+01,1.374039000e+00.
:FETCh:CNRatio?

:FETCh:CNRatio:CARRier?

Syntax
:FETCh:CNRatio:CARRier?

Description
Queries the carrier power.

Remarks
This command is only valid when the C/N ratio measurement is enabled in GPSA mode.
The power unit of the returned value is consistent with the current Y-axis unit.

Return Format
The query returns the carrier power in scientific notation.

Example
The following query returns -1.484203000e+01.
:FETCh:CNRatio:CARRier?

:FETCh:CNRatio:CNRatio?

Syntax
:FETCh:CNRatio:CNRatio?

Description
Queries the results of C/N ratio measurement.
Remarks
This command is only valid when the C/N ratio measurement is enabled in GPSA mode.

Return Format
The query returns the C/N ratio in scientific notation.

Example
The following query returns 8.956909000e-02.
:FETCh:CNRatio:CNRatio?

:FETCh:CNRatio:NOISe?

Syntax
:FETCh:CNRatio:NOISe?

Description
Queries the noise power.

Remarks
This command is only valid when the C/N ratio measurement is enabled in GPSA mode.
The power unit of the returned value is consistent with the current Y-axis unit.

Return Format
The query returns the noise power in scientific notation.

Example
The following query returns -1.442294000e+01.
:FETCh:CNRatio:NOISe?

:FETCh:ACPower?

Syntax
:FETCh:ACPower?

Description
Queries the results of adjacent channel power measurement.

Remarks
This command is only valid when the adjacent channel power measurement is enabled in GPSA mode.
The power unit of the returned value is consistent with the current Y-axis unit.

Return Format
The query returns 5 values (main channel power, upper channel power, the power difference between the upper channel and the main channel (in dBc), the lower channel power, the power difference between the lower channel and the main channel (in dBc)) in scientific notation, and the values are separated by commas.

Example
The following query returns -5.150423000e+01,-5.173441000e+01,-2.301865000e-01,-5.142665000e+01,7.757568000e-02.
:FETCh:ACPower?

:FETCh:ACPower:LOWer?

Syntax
:FETCh:ACPower:LOWer?

Description
Queries the lower channel power of the adjacent channel power measurement.

Remarks
This command is only valid when the adjacent channel power measurement is enabled in GPSA mode.
The power unit of the returned value is consistent with the current Y-axis unit.

Return Format
The query returns the lower channel power in scientific notation.

Example
The following query returns -5.142665000e+01.
:FETCh:ACPower:LOWer?

:FETCh:ACPower:MAIN?

Syntax
:FETCh:ACPower:MAIN?

Description
Queries the main channel power of the adjacent channel power measurement.

Remarks
This command is only valid when the adjacent channel power measurement is enabled in GPSA mode.
The power unit of the returned value is consistent with the current Y-axis unit.

Return Format
The query returns the main channel power in scientific notation.

Example
The following query returns -5.150423000e+01.
:FETCh:ACPower:MAIN?

:FETCh:ACPower:UPPer?

Syntax
:FETCh:ACPower:UPPer?

Description
Queries the upper channel power of the adjacent channel power measurement.

Remarks
This command is only valid when the adjacent channel power measurement is enabled in GPSA mode.
The power unit of the returned value is consistent with the current Y-axis unit.

Return Format
The query returns the upper channel power in scientific notation.

Example
The following query returns -5.173441000e+01.
:FETCh:ACPower:UPPer?

:FETCh:EBWidth?

Syntax
:FETCh:EBWidth?

Description
Queries the result of the emission bandwidth measurement.

Remarks
This command is only valid when the occupied bandwidth measurement is enabled in GPSA mode.

Return Format
The query returns the emission bandwidth in integer, and its unit is Hz.

Example
The following query returns 5.000000000e+04.
:FETCh:EBWidth?

:FETCh:HARMonics:AMPLitude:ALL?

Syntax
:FETCh:HARMonics:AMPLitude:ALL?

Description
Queries the amplitudes of the first 10 harmonics. The first harmonic is the fundamental waveform.

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.
The amplitude unit of the return value is consistent with the current Y-axis unit.

Return Format
The query returns the amplitudes of the first 10 harmonics in scientific notation. They are separated by commas.
If the number of harmonics under measurement is less than 10, the harmonics that are not measured do not have return values.

Example
The following query returns -1.692102000e+01,-6.458423000e+01,-7.509421000e+01,-7.924328000e+01,-7.847027000e+01,-7.885457000e+01,-7.882358000e+01,-7.921457000e+01,-7.923057000e+01,-7.915358000e+01.
:FETCh:HARMonics:AMPLitude:ALL?

:FETCh:HARMonics:AMPLitude<n>?

Syntax
:FETCh:HARMonics:AMPLitude<n>?

Description
Queries the amplitude of the specified harmonic.

Parameter

Name	Type	Range	Default
<n>	Integer	1 to 10	——

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.
The amplitude unit of the return value is consistent with the current Y-axis unit.
The query returns --- when the harmonic read does not have data.

Return Format
The query returns the amplitude of the specified harmonic in scientific notation.

Example
The following query returns -1.692102000e+01.
:FETCh:HARMonics:AMPLitude1?

:FETCh:HARMonics[:DISTortion]?

Syntax
:FETCh:HARMonics[:DISTortion]?

Description
Queries the percentage of the total harmonic distortion.

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.

Return Format
The query returns the percentage of the total harmonic distortion in scientific notation.

Example
The following query returns 1.510000000e-02.
:FETCh:HARMonics:DISTortion?

:FETCh:HARMonics:FREQuency:ALL?

Syntax
:FETCh:HARMonics:FREQuency:ALL?

Description
Queries the frequencies of the first 10 harmonics. The first harmonic is the fundamental waveform.

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.

Return Format
The query returns the frequencies of the first 10 harmonics in integer (separated by commas). The unit is Hz. If the number of harmonics under measurement is less than 10, the harmonics that are not measured do not have return values.

Example
The following query returns
4.550000000e+07,9.100000000e+07,1.365000000e+08,1.820000000e+08,2.275000000e+08,2.730000000e+08,3.185000000e+08,3.640000000e+08,4.095000000e+08,4.550000000e+08.
:FETCh:HARMonics:FREQuency:ALL?

:FETCh:HARMonics:FREQuency<n>?

Syntax
:FETCh:HARMonics:FREQuency<n>?

Description
Queries the frequency of the specified harmonic.

Parameter

Name	Type	Range	Default
<n>	Integer	1 to 10	——

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.
The query returns --- when the harmonic read does not have data.

Return Format
The query returns the frequency of the specified harmonic in scientific notation. The unit is Hz.

Example
The following query returns 4.550000000e+07.
:FETCh:HARMonics:FREQuency1?

:FETCh:HARMonics:FUNDamental?

Syntax
:FETCh:HARMonics:FUNDamental?

Description
Queries the frequency of the fundamental waveform.

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.

Return Format
The query returns the frequency of the fundamental waveform in scientific notation. The unit is Hz.

Example
The following query returns 4.550000000e+07.
:FETCh:HARMonics:FUNDamental?

:FETCh:OBWidth?

Syntax
:FETCh:OBWidth?

Description
Queries the results of the occupied bandwidth measurement.

Remarks
This command is only valid when the occupied bandwidth measurement is enabled in GPSA mode.

Return Format
The query returns the occupied bandwidth (Hz) and the transmit frequency error (Hz) in scientific notation. They are separated by commas.

Example
The following query returns 1.860000000e+06,2.000000000e+04.
:FETCh:OBWidth?

:FETCh:OBWidth:OBWidth?

Syntax
:FETCh:OBWidth:OBWidth?

Description
Queries the occupied bandwidth.

Remarks
This command is only valid when the occupied bandwidth measurement is enabled in GPSA mode.

Return Format
The query returns the occupied bandwidth (IBW) in scientific notation. Its unit is Hz.

Example
The following query returns 1.860000000e+06.
:FETCh:OBWidth:OBWidth?

:FETCh:OBWidth:OBWidth:FERRor?

Syntax
:FETCh:OBWidth:OBWidth:FERRor?

Description
Queries the transmit frequency error.

Remarks
This command is only valid when the occupied bandwidth measurement is enabled in GPSA mode.

Return Format
The query returns the transmit frequency error in scientific notation. The unit is Hz.

Example
The following query returns 2.000000000e+04.
:FETCh:OBWidth:OBWidth:FERRor?

:FETCh:SANalyzer<n>?

Syntax
:FETCh:SANalyzer<n>?

Description
Queries the trace data.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	——

Remarks
This command is only available for GPSA mode.

Return Format
The query returns the measurement data of the specified trace in scientific notation. The query returns the measurement data of the specified trace in scientific notation, separated by commas. Its unit is Hz.

Example
The following query returns 0,0,0,0,-1.000000000e+02,0,801,0,0,0,-3.19e+01,0.00e+00,0.00e+00,0.00e+00,0.00e+00,0.00e+00,0.00e+00,0.00e+00.
:FETCh:SANalyzer1?

:FETCh:TOIntercept?

Syntax
:FETCh:TOIntercept?

Description
Queries the results of TOI measurement.

Remarks
This command is only valid when the TOI measurement is enabled in GPSA mode.
The amplitude unit of the return value is consistent with the current Y-axis unit.

Return Format
The query returns the measurement results of TOI in the following format:
Base Lower frequency (Hz), amplitude, Base Upper frequency (Hz), amplitude, 3rd Order Lower frequency (Hz), amplitude, intercept, 3rd Order Upper frequency (Hz), amplitude, intercept.

Example
The following query returns 1.500450000e+09,-8.131735000e+01,1.500450000e+09,-8.131735000e+01,1.500450000e+09,-8.131735000e+01,-8.131735000e+01,1.500450000+09,-8.131735000e+01,-8.131735000e+01.
:FETCh:TOIntercept?

:FETCh:TOIntercept:IP3?

Syntax
:FETCh:TOIntercept:IP3?

Description
The query returns the minor value of the intercepts of the Third Order Lower and Third Order Upper.

Remarks
This command is only valid when the TOI measurement is enabled in GPSA mode.

Return Format
The query returns the minor one in scientific notation.

Example
The following query returns -8.131735000e+01.
:FETCh:TOIntercept:IP3?

:FETCh:TPOWer?

Syntax
:FETCh:TPOWer?

Description
Queries the result of T-power measurement.

Remarks
This command is only valid when the time-domain power measurement is enabled in GPSA mode.
The power unit of the returned value is consistent with the current Y-axis unit.

Return Format
The query returns the T-power measurement result in scientific notation.

Example
The following query returns -1.658941000e+01.
:FETCh:TPOWer?

:FETCh:CHPower:DENSity<n>?

Syntax
:FETCh:CHPower:DENSity<n>?

Description
Queries the channel power spectral density.

Parameter

Name	Type	Range	Default
<n>	Integer	1 to 10	——

Remarks
This command is only valid when the channel power measurement is enabled in GPSA mode.

Return Format
The query returns the power of the specified channel in scientific notation. The unit is Hz.

Example
The following query returns 4.550000000e+07.
:FETCh:CHPower:DENSity1?

:FETCh:CHPower<n>?

Syntax
:FETCh:CHPowe<n>?

Description
Queries the channel power.

Parameter

Name	Type	Range	Default
<n>	Integer	1 to 10	——

Remarks
This command is only valid when the channel power measurement is enabled in GPSA mode.

Return Format
The query returns the power of the specified channel in scientific notation.

Example
The following query returns 4.550000000e+07.
:FETCh:CHPower:DENSity1?

IEEE 488.2 Common Commands

IEEE 488.2 common commands are used to operate or query the status registers.

*CLS

Syntax
*CLS

Description
Clears all the event registers and status byte registers.

*ESE

Syntax
*ESE <value>
*ESE?

Description
Sets the enable register for the standard event status register.
Queries the enable register for the standard event status register.

Parameter

Name	Type	Range	Default
<value>	Integer	Refer to "**Remarks**"	0

Remarks
The bit 2, bit 3, bit 4, and bit 7 are reserved; you can set their values but they will not affect the system. Bit 1 and bit 6 are not used and are always treated as 0; therefore, the range of <value> are the decimal numbers corresponding to the binary numbers ranging from 00000000 (0 in decimal) to 11111111 (255 in decimal) and of which bit 1 and bit 6 are 0.

Return Format
The query returns an integer. The integer equals to the binary-weighted sum of all the bits set in the register. For example, the query returns 144 if Bit 4 (16 in decimal) and Bit 7 (128 in decimal) are enabled.

Example
The following command sets the enable register for the standard event status register to 16.
*ESE 16

The following query returns 16.
*ESE?

*ESR?

Syntax
*ESR?

Description
Queries and clears the event register for the standard event status register.

Remarks
Bit 1 and Bit 6 in the standard event status register are not in use, and are regarded as 0. The query returns a decimal value that corresponds to the binary values ranging from 00000000 (0 in decimal) to 11111111 (255 in decimal) and of which Bit 1 and Bit 6 are 0.

Return Format
The query returns an integer. The integer equals to the binary-weighted sum of all the bits set in the register. For example, the query returns 144 if Bit 4 (16 in decimal) and Bit 7 (128 in decimal) are enabled.

Example
The following query returns 24 (Bit 3 and Bit 4 have been set).
*ESR?

*IDN?

Syntax
*IDN?

Description
Queries the ID string of instrument.

Return Format
The query returns the ID string in the following format:
Rigol Technologies,<model>,<serial number>,XX.XX.XX
<model>: instrument model
<serial number>: serial number of the instrument
XX.XX.XX: software version of the instrument

Example
The following query returns Rigol Technologies,RSA814,RSA8004000000,00.00.25.
*IDN?

*OPC

Syntax
*OPC
*OPC?

Description
Sets Bit 0 (Operation Complete, OPC) in the standard event status register to 1 after the current operation is finished.
Queries whether the current operation is finished.

Return Format
The query returns 1 after the current operation is finished; otherwise, the query returns 0.

*RCL

Syntax
*RCL <integer>

Description
Recalls the selected register.

Parameter

Name	Type	Range	Default
<integer>	Integer	1 to 16	——

Example
The following command recalls Register 1.

*RCL 1

*RST

Syntax
*RST

Description
Restores the instrument to its factory default settings.

*SAV

Syntax
*SAV <integer>

Description
Saves the current instrument state to the selected register.

Parameter

Name	Type	Range	Default
<integer>	Integer	1 to 16	——

Example
The following command saves the current instrument state to Register 1.
*SAV 1

*SRE

Syntax
*SRE <value>
*SRE?

Description
Sets the enable register for the status byte register.
Queries the bits in the service request register.

Parameter

Name	Type	Range	Default
<value>	Integer	Refer to "**Remarks**"	0

Remarks
Bit 0 and Bit 1 are not used and are always treated as 0; therefore, the range of <value> are the decimal numbers corresponding to the binary numbers ranging from 00000000 (0 in decimal) to 11111111 (255 in decimal) and of which Bit 0 and Bit 1 are 0.

Return Format
The query returns an integer. The integer equals to the binary-weighted sum of all the bits set in the register. For example, the query returns 144 if Bit 4 (16 in decimal) and Bit 7 (128 in decimal) are enabled.

Example
The following command sets the enable register for the status byte register to 16.
*SRE 16
The following query returns 16.
*SRE?

*STB?

Syntax
*STB?

Description
Queries the event register for the status byte register.

Remarks
Bit 0 and Bit 1 in the status byte register are not in use, and are regarded as 0. The query returns a decimal value that corresponds to the binary values ranging from 00000000 (0 in decimal) to 11111111 (255 in decimal) and of which Bit 0 and Bit 1 are 0.

Return Format
The query returns an integer. The integer equals to the binary-weighted sum of all the bits set in the register. For example, the query returns 144 if Bit 4 (16 in decimal) and Bit 7 (128 in decimal) are enabled.

Example
The following query returns 24 (Bit 3 and Bit 4 have been set).
*STB?

*TST?

Syntax
*TST?

Description
Queries whether the self-check operation is finished.

Remarks
The query returns 0 or 1. A zero is returned if the test is successful, 1 if it fails.

*WAI

Syntax
*WAI

Description
Waits for all the pending operations to complete before executing any additional commands.

:GPIB:PARSe:END

Syntax
:GPIB:PARSe:END

Description
The GPIB module command. This instruction set sends the "Parse End" command to the GPIB module. If this command is not set, the USB-GPIB adapter module fails to work. You will receive no return value after sending a command.

:INITiate Commands

:INITiate:CONTinuous

Syntax
:INITiate:CONTinuous <bool>
:INITiate:CONTinuous?

Description
Selects continuous (ON|1) or single (OFF|0) measurement mode.
Queries the current measurement mode.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Return Format
The query returns 0 or 1.

Example
The following command sets the instrument to sweep continuously.
:INITiate:CONTinuous ON or :INITiate:CONTinuous 1

The following query returns 1.
:INITiate:CONTinuous?

:INITiate[:IMMediate]

Syntax
:INITiate[:IMMediate]

Description
Initializes one sweep in the non-measurement state.
Triggers one measurement in the measurement state.

:INITiate:RESTart

Syntax
:INITiate:RESTart

Description
Restarts the sweep.

:INPut Commands

:INPut:IF:OVERload?

Syntax
:INPut:IF:OVERload?

Description
Queries the IF overload.

Return Format
The query returns 0 or 1.
0 indicates not overloaded. 1 indicates overloaded.

:INSTrument Commands

:INSTrument:COUPle:FREQuency:CENTer

Syntax
:INSTrument:COUPle:FREQuency:CENTer <enum>
:INSTrument:COUPle:FREQuency:CENTer?

Description
Sets the setting status of the global center frequency of the instrument.
Queries the setting status of the global center frequency of the instrument.

Parameter

Name	Type	Range	Default
<enum>	Discrete	ALL|NONE	NONE

Remarks
NONE: turns off the global center frequency.
ALL: turns on the global center frequency.
If you execute this command in any mode, the center frequency of the current mode is set to the global center frequency. Adjusting the center frequency in a mode, while the global center frequency is on, will modify the global center frequency.

Return Format
The query returns ALL or NONE.

Example
The following command enables the global center frequency of the instrument.
:INSTrument:COUPle:FREQuency:CENTer ALL

The following query returns ALL.
:INSTrument:COUPle:FREQuency:CENTer?

:INSTrument:SCReen:CATalog?

Syntax
:INSTrument:SCReen:CATalog?

Description
Queries the available measurement modes displayed at the bottom of the screen.

:INSTrument:SCReen:CREate

Syntax
:INSTrument:SCReen:CREate

Description
Creates a new measurement mode label at the bottom of the screen.

:INSTrument:SCReen:DELete

Syntax
:INSTrument:SCReen:DELete

Description
Deletes the currently selected measurement mode label at the bottom of the screen.

:INSTrument:SCReen:DELete:ALL

Syntax
:INSTrument:SCReen:DELete:ALL

Description
Deletes all the currently not selected measurement mode label at the bottom of the screen.

:INSTrument:SCReen:REName

Syntax
:INSTrument:SCReen:REName<name>

Description
Renames the currently selected measurement mode label at the bottom of the screen.

Parameter

Name	Type	Range	Default
<name>	ASCII String	The label can contain English letters and numbers, as well as some symbols.	-

:INSTrument:SCReen:SELect

Syntax
:INSTrument:SCReen:SELect <name>
:INSTrument:SCReen:SELect?

Description
Selects the specified measurement mode label at the bottom of the screen.
Queries the measurement mode label at the bottom of the screen that you select currently.

Parameter

Name	Type	Range	Default
<name>	ASCII String	The label can contain English letters and numbers, as well as some symbols.	-

Remarks
You can run the *:INSTrument:SCReen:CATalog?* command to query the measurement mode label name.

Example
The following command sets to select measurement mode label RTSA 1 at the bottom of the screen.
:INSTrument:SCReen:SELect RTSA 1
The following query returns RTSA 1.
:INSTrument:SCReen:SELect?

:INSTrument[:SELect]

Syntax
:INSTrument[:SELect] <enum>
:INSTrument[:SELect]?

Description
Selects the working mode of the instrument.
Queries the working mode of the instrument.

Parameter

Name	Type	Range	Default
<enum>	Discrete	SA|GPSA_TG|RTSA| RTSA_UFS|GPSA_ACP| GPSA_CNR|GPSA_CHPower|GPSA_TOI|GPSA_HARModist| GPSA_OBW|GPSA_TPOWer|EMI|VMA|AM|FM|PM	SA

Remarks
After running the command of switching the working mode, we recommend you set the timeout value to 8 s, or perform the next operation after a delay of 8 s.

Example
The following command sets the working mode of the instrument to GPSA.
:INSTrument:SELect SA

The following query returns 1 or SA.
:INSTrument:SELect?

:MMEMory Commands

:MMEMory:DELete

Syntax
:MMEMory:DELete <file_name>

Description
Deletes a specified file.

Parameter

Name	Type	Range	Default
<file_name>	ASCII String	——	--

Remarks
<file_name> should contain the path and the filename.
This operation fails if the file with the specified filename does not exist.

Example
The following command deletes the "state1.sta" file from the "/ GPSA/state" folder.
:MMEMory:DELete /GPSA/state/state1.sta

:MMEMory:LOAD:FMT

Syntax
:MMEMory:LOAD:FMT <label>,<file_name>

Description
Loads the edited FMT file (.csv).

Parameter

Name	Type	Range	Default
<label>	Discrete	UPPer|LOWer	——
<file_name>	ASCII String	——	--

Remarks
This operation fails if the file with the specified filename does not exist.
This command is only available for RTSA mode.

Example
The following command loads the FMT file (mask1.csv) to the upper mask.
:MMEMory:LOAD:FMT UPPer,mask1.csv

:MMEMory:LOAD:LIMit

Syntax
:MMEMory:LOAD:LIMit <label>,<file_name>

Description
Loads the edited limit line file (.csv).

Parameter

Name	Type	Range	Default
<label>	Discrete	LLINE1|LLINE2|LLINE3|LLINE4|LLINE5|LLINE6	——
<file_name>	ASCII String	——	--

Remarks
This operation fails if the file with the specified filename does not exist.

Example
The following command loads the limit line file (upp1.csv) to Limit1.
:MMEMory:LOAD:LIMit LLINE1,upp1.csv

:MMEMory:LOAD:CORRection

Syntax
:MMEMory:LOAD:CORRection <label>,<file_name>

Description
Loads the amplitude correction data (.csv).

Parameter

Name	Type	Range	Default
<label>	Discrete	LLINE1|LLINE2|LLINE3|LLINE4|LLINE5|LLINE6	——
<file_name>	ASCII String	——	--

Remarks
This operation fails if the file with the specified filename does not exist.

Example
The following command loads the amplitude correction data file (upp1.csv) to Limit1.
:MMEMory:LOAD:CORRection LLINE1,upp1.csv

:MMEMory:LOAD:STATe

Syntax
:MMEMory:LOAD:STATe <file_name>

Description
Loads the specified state file (.sta).

Parameter

Name	Type	Range	Default
<file_name>	ASCII String	——	--

Remarks
This operation fails if the file with the specified filename does not exist.

Example
The following command loads the state file (state1.sta) to the instrument.
:MMEMory:LOAD:STATe state1.sta

:MMEMory:LOAD:TRACe:DATA

Syntax
:MMEMory:LOAD:TRACe:DATA <label>,<file_name>

Description
Loads the specified measurement data file (suffixed with .csv).

Parameter

Name	Type	Range	Default
<label>	Discrete	TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6	——
<file_name>	ASCII String	——	--

Remarks
This operation fails if the file with the specified filename does not exist.

Example
The following command loads the measurement data file (trace1.csv) to Trace1.
:MMEMory:LOAD:TRACe:DATA TRACE1,trace1.csv

:MMEMory:LOAD:RESults

Syntax
:MMEMory:LOAD:RESults <label>,<path>

Description
Loads the measurement results.

Parameter

Name	Type	Range	Default
<label>	Discrete	TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6| PTABLE|MTABLE|RESULT|RAWDATA	——
<path>	Real	——	--

Remarks
This operation fails if the file with the specified filename does not exist.

Example
The following command loads the measurement data file (trace1.csv) to Trace1.
:MMEMory:LOAD:RESults TRACE1,trace1.csv

:MMEMory:MOVE

Syntax
:MMEMory:MOVE <file_name1>,<file_name2>

Description
Renames the specified file <file_name1> as <file_name2>.

Parameter

Name	Type	Range	Default
<file_name1>	ASCII String	——	--
<file_name2>	ASCII String	——	--

Remarks
<file_name1> and <file_name2> should contain the path and the filename.
This operation fails if the file with the specified filename does not exist.

Example
The following command renames the state file (state1.sta) in the folder (/GPSA/state) as "state2.sta".
:MMEMory:MOVE /GPSA/state/state1.sta,/GPSA/state/state2.sta

:MMEMory:STORe:IMG:PNG

Syntax
:MMEMory:STORe:IMG:PNG <path>

Description
Saves the screen image to the specified path.

Parameter

Name	Type	Range	Default
<path>	ASCII String	——	——

:MMEMory:STORe:STATe

Syntax
:MMEMory:STORe:STATe <file_name>

Description
Saves the current instrument state with the specified filename suffixed with ".sta" to the default path (/"mode name"/state).

Parameter

Name	Type	Range	Default
<file_name>	ASCII String	——	--

Remarks
If the specified file already exists, overwrite it.

Example
The following command saves the current instrument state with the filename "state.sta" to the folder (/GPSA/state).
:MMEMory:STORe:STATe state

:MMEMory:STORe:TRACe

Syntax
:MMEMory:STORe:TRACe <label>,<file_name>

Description
Saves the specified trace+state file with filename suffixed with ".trs" to the default path (/"mode name"/tracestate).

Parameter

Name	Type	Range	Default
<label>	Discrete	TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6	——
<file_name>	ASCII String	——	——

Remarks
If the specified file already exists, overwrite it.

Example
The following command saves the current instrument state and Trace 1 with the filename "mystate.trs" to the folder (/GPSA/tracestate).
:MMEMory:STORe:TRACe TRACE1,mystate

:MMEMory:STORe:TRACe:DATA

Syntax
:MMEMory:STORe:TRACe:DATA <label>,<file_name>

Description
Saves the trace measurement data with a specified filename suffixed with ".csv" to the default path (/"mode name"/measdata).

Parameter

Name	Type	Range	Default
<label>	Discrete	TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6	——
<file_name>	ASCII String	——	——

Remarks
If the specified file already exists, overwrite it.

Example
The following command saves Trace1 measurement data with the specified filename "mydata.csv" to the folder (/GPSA/measdata).
:MMEMory:STORe:TRACe:DATA TRACE1,mydata

:MMEMory:STORe:RESults

Syntax
:MMEMory:STORe:RESults <file_name>

Description
Saves the current measurement results with a specified filename suffixed with ".csv" by default (you do not have to add the suffix manually) to the default path (/"mode name"/measdata).

Parameter

Name	Type	Range	Default
<file_name>	ASCII String	——	——

Remarks
If the specified file already exists, overwrite it.
This command is only valid when the specified advanced measurement is enabled in GPSA mode.

Example
The following command saves the current measurement results with the specified filename "data" to the folder (/GPSA/measdata).
:MMEMory:STORe:RESults data

:MMEMory:STORe:MTABle

Syntax
:MMEMory:STORe:MTABle <file_name>

Description
Saves the marker table with a specified filename suffixed with ".csv" by default (you do not have to add the suffix manually) to the default path (/"mode name"/measdata).

Parameter

Name	Type	Range	Default
<file_name>	ASCII String	——	——

Remarks
If the specified file already exists, overwrite it.

Example
The following command saves the marker table with the specified filename "MAK1" to the folder (/GPSA/measdata).
:MMEMory:STORe:MTABle MAK1

:MMEMory:STORe:PTABle

Syntax
:MMEMory:STORe:PTABle <file_name>

Description
Saves the peak table with a specified filename suffixed with ".csv" by default (you do not have to add the suffix manually) to the default path (/"mode name"/measdata).

Parameter

Name	Type	Range	Default
<file_name>	ASCII String	——	——

Remarks
If the specified file already exists, overwrite it.

Example
The following command saves the peak table with the specified filename "PT1" to the folder (/GPSA/measdata).
:MMEMory:STORe:PTABle PT1

:MMEMory:STORe:LIMit

Syntax
:MMEMory:STORe:LIMit <label>,<file_name>

Description
Saves the currently edited limit line with a specified filename suffixed with ".csv" by default to the default path (/"mode name"/limit).

Parameter

Name	Type	Range	Default
<label>	Discrete	LLINE1|LLINE2|LLINE3|LLINE4|LLINE5|LLINE6	——
<file_name>	ASCII String	——	——

Remarks
If the specified file already exists, overwrite it.

Example
The following command saves Limit1 data with the filename "low" to the folder (/GPSA/limit).
:MMEMory:STORe:LIMit LLINE1,low

:MMEMory:STORe:CORRection

Syntax
:MMEMory:STORe:CORRection <label>,<file_name>

Description
Saves the amplitude correction data of the current limit line with a specified filename suffixed with ".csv" by default (you do not have to add the suffix manually) to the default path (/"mode name"/limit).

Parameter

Name	Type	Range	Default
<label>	Discrete	LLINE1|LLINE2|LLINE3|LLINE4|LLINE5|LLINE6	——
<file_name>	ASCII String	——	——

Remarks
If the specified file already exists, overwrite it.

Example
The following command saves amplitude correction data of Limit1 with the filename "low" to the folder (/GPSA/limit).
:MMEMory:STORe:CORRection LLINE1,low

:MMEMory:STORe:FMT

Syntax
:MMEMory:STORe:FMT <label>,<file_name>

Description
Saves the FMT data of the current limit line with a specified filename suffixed with ".csv" by default (you do not have to add the suffix manually) to the default path (/"mode name"/limit).

Parameter

Name	Type	Range	Default
<label>	Discrete	LLINE1|LLINE2|LLINE3|LLINE4|LLINE5|LLINE6	——
<file_name>	ASCII String	——	——

Remarks
If the specified file already exists, overwrite it.

Example
The following command saves FMT data with the filename "low" to the folder (/GPSA/limit).
:MMEMory:STORe:FMT LLINE1,low

:MMEMory:STORe:SIGNal

Syntax
:MMEMory:STORe:SIGNal <file_name>

Description
Saves the signal with a specified filename suffixed with ".csv" by default (you do not have to add the suffix manually) to the default path (/"mode name"/measdata).

Parameter

Name	Type	Range	Default
<file_name>	ASCII String	——	——

Remarks
If the specified file already exists, overwrite it.

Example

The following command saves the signal with the specified filename "PT1" to the folder (/GPSA/measdata).
:MMEMory:STORe:SIGNal PT1

:MMEMory:STORe:RESults

Syntax
:MMEMory:STORe:RESults <file_name>

Description
Saves the current measurement results with a specified filename suffixed with ".csv" by default (you do not have to add the suffix manually) to the default path (/"mode name"/measdata).

Parameter

Name	Type	Range	Default
<file_name>	ASCII String	——	——

Remarks
If the specified file already exists, overwrite it.
This command is only valid when the specified advanced measurement is enabled in GPSA mode.

Example
The following command saves the current measurement results with the specified filename "data" to the folder (/GPSA/measdata).
:MMEMory:STORe:RESults data

:MMEMory:STORe:RETurn?

Syntax
:MMEMory:STORe:RETurn?

Description
Saves the returned results.

:MMEMory:STORe:SCReen

Syntax
:MMEMory:STORe:SCReen <file>

Description
Saves the current screen image with the specified filename suffixed with "*.jpg", "*.png/", or "*.bmp" to the default path (/"mode name"/screen).

Parameter

Name	Type	Range	Default
<file>	ASCII String	——	——

Remarks
If the specified file already exists, overwrite it.
If a suffix (.jpg/.png/.bmp) is added to the filename, you can save the current screen image with a different format based on its different suffix.
If no suffix is added to the filename, then by default, the current screen image is saved in the currently selected format.

Example

The following command saves the current screen image with the filename "screen.jpg" to the folder (/GPSA/screen).
:MMEMory:STORe:SCReen screen.jpg

:MMEMory:STORe:SCReen:DATA?

Syntax
:MMEMory:STORe:SCReen:DATA?

Description
Saves the screen data.

:MMEMory:USER:STORe

Syntax
:MMEMory:USER:STORe <user>
:MMEMory:USER:STORe?

Description
Sets the preset settings.
Queries the preset settings.

Parameter

Name	Type	Range	Default
<user>	Discrete	DEF|USER1|USER2|USER3|USER4|USER5|USER6	DEF

Remarks
N/A

Example
The following command selects USER1.
:MMEMory:USER:STORe USER1

The following query command returns USER1.
:MMEMory:USER:STORe?

:OUTPut Commands

:OUTPut[:EXTernal][:STATe]

Syntax
:OUTPut[:EXTernal][:STATe] <bool>
:OUTPut[:EXTernal][:STATe]?

Description
Enables or disables the output of the TG.
Queries the on/off status of the tracking generator output.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only available in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the output of the TG.
:OUTPut:EXTernal:STATe ON or :OUTPut:EXTernal:STATe 1

The following query returns 1.
:OUTPut:EXTernal:STATe?

:OUTPut:CW[:STATe]

Syntax
:OUTPut:CW[:STATe] <bool>
:OUTPut:CW[:STATe]?

Description
Enables or disables the standalone source output.
Queries the enable/disable status of the standalone source.

Parameter

Name	Type	Range	Default
<bool>	Boolean	OFF|ON|0|1	OFF|0

Remarks

This switch is only valid when the output source mode is set to Standalone Source. It is mutually exclusive with the tracking source switch, meaning the two switches cannot be enabled at the same time.
The standalone source switch will be automatically disabled once the output source mode is switched to Tracking Source.

Return Format
The query returns 0 or 1.

Example
The following command enables the standalone source output:

:OUTPu:CW:STATe ON or :OUTPut:CW:STATe 1

The following query returns 1.
:OUTPut:CW:STATe?

[:SENSe] Commands

[:SENSe]:ACPower:AVERage:COUNt

Syntax
[:SENSe]:ACPower:AVERage:COUNt <integer>
[:SENSe]:ACPower:AVERage:COUNt?

Description
Sets the average count of the ACP measurement.
Queries the average count of the ACP measurement.

Parameter

Name	Type	Range	Default
<integer>	Integer	1 to 1000	10

Remarks
This command is only valid when the adjacent channel power measurement is enabled in GPSA mode.

Return Format
The query returns the average count in integer.

Example
The following command sets the average count to 100.
:SENSe:ACPower:AVERage:COUNt 100

The following query returns 100.
:SENSe:ACPower:AVERage:COUNt?

[:SENSe]:ACPower:AVERage[:STATe]

Syntax
[:SENSe]:ACPower:AVERage[:STATe] <bool>
[:SENSe]:ACPower:AVERage[:STATe]?

Description
Enables or disables the average measurement function of the ACP measurement.
Queries the status of the average measurement function of the ACP measurement.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only valid when the adjacent channel power measurement is enabled in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the average measurement function.
:SENSe:ACPower:AVERage:STATe ON or :SENSe:ACPower:AVERage:STATe 1

The following query returns 1.
:SENSe:ACPower:AVERage:STATe?

[:SENSe]:ACPower:AVERage:TCONtrol

Syntax
[:SENSe]:ACPower:AVERage:TCONtrol <enum>
[:SENSe]:ACPower:AVERage:TCONtrol?

Description
Selects the average mode of the adjacent channel power measurement.
Queries the average mode of the adjacent channel power measurement.

Parameter

Name	Type	Range	Default
<enum>	Discrete	EXPonential|REPeat	EXPonential

Remarks
EXPonential: indicates the exponential average.
REPeat: indicates the repeat average.
When "EXPonential" is selected, the result is the exponential average of the measurement results obtained in the past N times (N is specified in *[:SENSe]:ACPower:AVERage:COUNt*).
When "REPeat" is selected, the result is the arithmetic average of the measurement results obtained in the past N times (N is specified in *[:SENSe]:ACPower:AVERage:COUNt*).
This command is only valid when the adjacent channel power measurement is enabled in GPSA mode.

Return Format
The query returns EXP or REP.

Example
The following command sets the average mode to Repeat.
:SENSe:ACPower:AVERage:TCONtrol REPeat

The following query returns REP.
:SENSe:ACPower:AVERage:TCONtrol?

[:SENSe]:ACPower:BANDwidth:ACHannel

Syntax
[:SENSe]:ACPower:BANDwidth:ACHannel <freq>
[:SENSe]:ACPower:BANDwidth:ACHannel?

Description
Sets the bandwidth of the adjacent channel.
Queries the bandwidth of the adjacent channel.

Parameter

Name	Type	Range	Default
<freq>	Real	3 Hz to 2.166666 GHz	2 MHz

Remarks
This command is only valid when the adjacent channel power measurement is enabled in GPSA mode.

Return Format
The query returns the bandwidth of the adjacent channel in scientific notation. Its unit is Hz.

Example
The following command sets the bandwidth of the adjacent channel to 1 MHz.
:SENSe:ACPower:BANDwidth:ACHannel 1000000

The following query returns 1.000000000e+06.
:SENSe:ACPower:BANDwidth:ACHannel?

[:SENSe]:ACPower:BANDwidth:INTegration

Syntax
[:SENSe]:ACPower:BANDwidth:INTegration <freq>
[:SENSe]:ACPower:BANDwidth:INTegration?

Description
Sets the bandwidth of the main channel.
Queries the bandwidth of the main channel.

Parameter

Name	Type	Range	Default
<freq>	Real	3 Hz to 2.166666 GHz	2 MHz

Remarks
This command is only valid when the adjacent channel power measurement is enabled in GPSA mode.

Return Format
The query returns the bandwidth of the main channel in scientific notation. Its unit is Hz.

Example
The following command sets the bandwidth of the main channel to 1 MHz.
:SENSe:ACPower:BANDwidth:INTegration 1000000

The following query returns 1.000000000e+06.
:SENSe:ACPower:BANDwidth:INTegration?

[:SENSe]:ACPower:CSPacing

Syntax
[:SENSe]:ACPower:CSPacing <freq>
[:SENSe]:ACPower:CSPacing?

Description
Sets the center frequency difference (channel spacing) between the main channel and the adjacent channels.
Queries the channel spacing.

Parameter

Name	Type	Range	Default
<freq>	Real	3 Hz to 2.166666 GHz	2 MHz

Remarks
This command is only valid when the adjacent channel power measurement is enabled in GPSA mode.

Return Format
The query returns the channel spacing in scientific notation. Its unit is Hz.

Example
The following command sets the channel spacing to 1 MHz.
:SENSe:ACPower:CSPacing 1000000

The following query returns 1.000000000e+06.
:SENSe:ACPower:CSPacing?

[:SENSe]:ACQuisition:TIME

Syntax
[:SENSe]:ACQuisition:TIME <time>
[:SENSe]:ACQuisition:TIME?

Description
Sets the RTSA acquisition time. The unit is s.
Queries the RTSA acquisition time.

Parameter

Name	Type	Range	Default
<time>	Real	32 ms to 40 s (Density) 100 μs to 40 s (Other)	1 ms

Remarks
This command is only available for Normal view, Density view, Spectrogram view, and Density Spectrogram view in RTSA mode.

Return Format
The query returns the acquisition time in scientific notation.

Example
The following command sets the acquisition time to 0.5 s.
:SENSe:ACQuisition:TIME 0.5

The following query returns 5.000000000e-01.
:SENSe:ACQuisition:TIME?

[:SENSe]:ACQuisition:TIME:AUTO

Syntax
[:SENSe]:ACQuisition:TIME:AUTO <bool>
[:SENSe]:ACQuisition:TIME:AUTO?

Description
Sets the RTSA acquisition time auto state.
Queries the RTSA acquisition time auto state.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
This command is only available for Normal view, Density view, Spectrogram view, and Density Spectrogram view in RTSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the auto acquisition time for producing one single trace or one bitmap.
:SENSe:ACQuisition:TIME:AUTO ON or :SENSe:ACQuisition:TIME:AUTO 1

The following query returns 1.
:SENSe:ACQuisition:TIME:AUTO?

[:SENSe]:ACQuisition:TIME:PVTime

Syntax
[:SENSe]:ACQuisition:TIME:PVTime <time>
[:SENSe]:ACQuisition:TIME:PVTime?

Description
Sets the RTSA PvT acquisition time. The unit is s.
Queries the RTSA PvT acquisition time.

Parameter

Name	Type	Range	Default
<time>	Real	0 s to 40 s	30 ms

Remarks
This command is only available for PvT view, PvT Spectrum view, and PvT Spectrogram view in RTSA mode.

Return Format
The query returns the acquisition time in scientific notation.

Example
The following command sets the acquisition time to 0.5 s.
:SENSe:ACQuisition:TIME:PVTime 0.5

The following query returns 5.000000000e-01.
:SENSe:ACQuisition:TIME:PVTime?

[:SENSe]:ACQuisition:TIME:PVTime:AUTO

Syntax
[:SENSe]:ACQuisition:TIME:PVTime:AUTO <bool>
[:SENSe]:ACQuisition:TIME:PVTime:AUTO?

Description
Sets the RTSA PvT acquisition time auto state.
Queries the RTSA PvT acquisition time auto state.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
This command is only available for PvT view, PvT Spectrum view, and PvT Spectrogram view in RTSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the auto acquisition time for producing all the traces.
:SENSe:ACQuisition:TIME:PVTime:AUTO ON or :SENSe:ACQuisition:TIME:PVTime:AUTO 1

The following query returns 1.
:SENSe:ACQuisition:TIME:PVTime:AUTO?

[:SENSe]:ACQuisition:POINts

Syntax
[:SENSe]:ACQuisition:POINts <value>
[:SENSe]:ACQuisition:POINts?

Description
Sets the number of sweep points.
Queries the number of sweep points.

Parameter

Name	Type	Range	Default
<value>	Integer	101 to 100001	801

Remarks
N/A

Return Format
The query returns the number of sweep points in integer.

Example
The following command sets the number of sweep points to 500.
:SENSe:ACQuisition:POINts 500

The following query returns 500.
:SENSe:ACQuisition:POINts?

[:SENSe]:ACQuisition:TIME

Syntax
[:SENSe]:ACQuisition:TIME <value>
[:SENSe]:ACQuisition:TIME?

Description
Sets the sweep time.
Queries the sweep time.

Parameter

Name	Type	Range	Default
<value>	Real	1 ms to 4 ks	151 ms

Remarks
N/A

Return Format
The query returns the sweep time in scientific notation.

Example
The following command sets the sweep time to 10 s.
:SENSe:ACQuisition:POINts 10

The following query returns 1000000000e+01.
:SENSe:ACQuisition:POINts?

[:SENSe]:ACQuisition:COUPle:AUTO

Syntax
[:SENSe]:ACQuisition:COUPle:AUTO

Description
Performs auto coupling.

[:SENSe]:AVERage:COUNt

Syntax
[:SENSe]:AVERage:COUNt <integer>
[:SENSe]:AVERage:COUNt?

Description
Sets the trace average count of the current measurement.
Queries the trace average count of the current measurement.

Parameter

Name	Type	Range	Default
<integer>	Integer	1 to 10000	100

Remarks
This command is applicable to swept SA in GPSA mode and RTSA mode.

Return Format
The query returns the average count in integer.

Example
The following command sets the average count to 100.
:SENSe:AVERage:COUNt 100

The following query returns 100.
:SENSe:AVERage:COUNt?

[:SENSe]:AVERage:CLEar

Syntax
[:SENSe]:AVERage:CLEar

Description
Clears the average count.

[:SENSe]:AVERage:COUNt:CURRent?

Syntax
[:SENSe]:AVERage:COUNt:CURRent?

Description
Queries the current average times of the average trace.

Return Format
The query returns the trace average count that has been executed in integer.

[:SENSe]:AVERage:TYPE

Syntax
[:SENSe]:AVERage:TYPE <enum>
[:SENSe]:AVERage:TYPE?

Description
Selects the average type of the swept SA analysis measurement.
Queries the average type of the swept SA analysis measurement.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	——
<enum>	Discrete	LOG|RMS|SCALar	LOG

Remarks
LOG: indicates log.
RMS: indicates root mean square.
SCALar: indicates scalar.
This command is only available for GPSA mode.

Return Format
The query returns LOG, RMS, or SCAL.

Example
The following command selects the average type of the swept SA analysis mode to Log.
:SENSe:AVERage:TYPE LOG

The following query returns LOG.
:SENSe:AVERage:TYPE?

[:SENSe]:AVERage:TYPE:AUTO

Syntax
[:SENSe]:AVERage:TYPE:AUTO <bool>
[:SENSe]:AVERage:TYPE:AUTO?

Description
Sets auto average state.
Queries auto average state.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only available for GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the auto state for the average type of the swept SA analysis mode.
:SENSe:AVERage:TYPE:AUTO ON or :SENSe:AVERage:TYPE:AUTO 1

The following query returns 1.
:SENSe:AVERage:TYPE:AUTO?

[:SENSe]:AVERage[:STATe]

Syntax
[:SENSe]:AVERage[:STATe] <bool>
[:SENSe]:AVERage[:STATe]?

Description
Sets the average state.
Queries the average state.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
The commands are only applicable to the instrument that has installed the advanced measurement option in GPSA.

Return Format
The query returns 0 or 1.

Example
The following command sets the average state to ON.
:SENSe:AVERage:STATe ON or :SENSe:AVERage:STATe 1

The following query returns 1.
:SENSe:AVERage:STATe?

[:SENSe]:AVERage:TCONtrol

Syntax
[:SENSe]:AVERage:TCONtrol <enum>
[:SENSe]:AVERage:TCONtrol?

Description
Sets the average mode of the swept SA measurement.
Queries the average mode of the swept SA measurement.

Parameter

Name	Type	Range	Default
<enum>	Discrete	EXPonential|REPeat	-

Remarks
EXPonential: indicates the exponential average mode.
REPeat: indicates the repeated average mode.

Return Format
The query returns EXP or REP.

Example
The following command sets the average mode of swept SA measurement to REPeat.
:SENSe:AVERage:TCONtrol REPeat

The following query returns REP.
:SENSe:AVERage:TCONtrol?

[:SENSe]:BANDwidth|BWIDth:EMIFilter:STATe

Syntax
[:SENSe]:BANDwidth|BWIDth:EMIFilter:STATe <bool>
[:SENSe]:BANDwidth|BWIDth:EMIFilter:STATe?

Description
Enables or disables the EMI filter.
Queries the on/off status of the EMI filter.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
ON: selects the EMI filter (-6 dB bandwidth).
OFF: selects the Gaussian filter (-3 dB bandwidth).
This command is only available for GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command selects the EMI filter.
:SENSe:BANDwidth:EMIFilter:STATe ON or :SENSe:BANDwidth:EMIFilter:STATe 1
:SENSe:BWIDth:EMIFilter:STATe ON or :SENSe:BWIDth:EMIFilter:STATe 1

The following query returns 1.
:SENSe:BANDwidth:EMIFilter:STATe?
:SENSe:BWIDth:EMIFilter:STATe?

[:SENSe]:BANDwidth|BWIDth[:RESolution]

Syntax
[:SENSe]:BANDwidth|BWIDth[:RESolution] <freq>
[:SENSe]:BANDwidth|BWIDth[:RESolution]?

Description
Sets the resolution bandwidth (RBW).
Queries the resolution bandwidth.

Parameter

Name	Type	Range	Default
<freq>	Discrete	1 Hz to 10 MHz (at 1-3-10 step)	3 MHz

Remarks
The setting command is only available for GPSA mode. The query command is available for both the GPSA mode and RTSA mode.

Return Format
In GPSA mode, the query returns the resolution bandwidth in scientific notation. Its unit is Hz.
In RTSA mode, the query returns the specific values of RBW1 through RBW6.

Example
The following command sets RBW to 1000 Hz.
:SENSe:BANDwidth:RESolution 1000
:SENSe:BWIDth:RESolution 1000

The following query returns 1.000000000e+03.
:SENSe:BANDwidth:RESolution?
:SENSe:BWIDth:RESolution?

[:SENSe]:BANDwidth|BWIDth[:RESolution]:AUTO

Syntax
[:SENSe]:BANDwidth|BWIDth[:RESolution]:AUTO <bool>
[:SENSe]:BANDwidth|BWIDth[:RESolution]:AUTO?

Description
Enables or disables the auto setting mode of RBW.
Queries the status of the auto setting mode of RBW.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
In auto mode, the resolution bandwidth changes with the span (non-zero span).
This command is only available for GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the auto setting mode of RBW.
:SENSe:BANDwidth:RESolution:AUTO ON or :SENSe:BANDwidth:RESolution:AUTO 1
:SENSe:BWIDth:RESolution:AUTO ON or :SENSe:BWIDth:RESolution:AUTO 1

The following query returns 1.
:SENSe:BANDwidth:RESolution:AUTO?
:SENSe:BWIDth:RESolution:AUTO?

[:SENSe]:BANDwidth|BWIDth[:RESolution]:SELect

Syntax
[:SENSe]:BANDwidth|BWIDth[:RESolution]:SELect <enum>
[:SENSe]:BANDwidth|BWIDth[:RESolution]:SELect?

Description
Sets the resolution bandwidth (RBW).
Queries the resolution bandwidth.

Parameter

Name	Type	Range	Default
<enum>	Discrete	RBW1|RBW2|RBW3|RBW4|RBW5|RBW6	RBW2

Remarks
This command is only available for RTSA mode. For details, refer to *RSA800 User Guide*.

Return Format
The query returns RBW1, RBW2, RBW3, RBW4, RBW5, or RBW6.

Example
The following command selects RBW1.

:SENSe:BANDwidth:RESolution:SELect RBW1
:SENSe:BWIDth:RESolution:SELect RBW1

The following query returns RBW1.
:SENSe:BANDwidth:RESolution:SELect?
:SENSe:BWIDth:RESolution:SELect?

[:SENSe]:BANDwidth|BWIDth[:RESolution]:SELect:AUTO

Syntax
[:SENSe]:BANDwidth|BWIDth[:RESolution]:SELect:AUTO <bool>
[:SENSe]:BANDwidth|BWIDth[:RESolution]:SELect:AUTO?

Description
Enables or disables the auto setting mode of RBW.
Queries the status of the auto setting mode of RBW.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
This command is only available for RTSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the auto setting mode of RBW.
:SENSe:BANDwidth:RESolution:SELect:AUTO ON or :SENSe:BANDwidth:RESolution:SELect:AUTO 1
:SENSe:BWIDth:RESolution:SELect:AUTO ON or :SENSe:BWIDth:RESolution:SELect:AUTO 1

The following query returns 1.
:SENSe:BANDwidth:RESolution:SELect:AUTO?
:SENSe:BWIDth:RESolution:SELect:AUTO?

[:SENSe]:BANDwidth|BWIDth:SHAPe

Syntax
[:SENSe]:BANDwidth|BWIDth:SHAPe <enum>
[:SENSe]:BANDwidth|BWIDth:SHAPe?

Description
Sets the filter type.
Queries the filter type.

Parameter

Name	Type	Range	Default
<enum>	Discrete	GAUSsian|FLATtop|BHARris|RECTangular|HANNing|KAISer	GAUSsian

Remarks
This command is only available for RTSA mode.
When "Rectangular" is set to be the filter type, the analyzer automatically selects RBW1, and RBW2 through RBW6 is disabled.

Return Format
The query returns GAUS, FLAT, BHAR, RECT, HANN, or KAIS.

Example
The following command sets the filter type to Gaussian.
:SENSe:BANDwidth:SHAPe GAUSsian
:SENSe:BWIDth:SHAPe GAUSsian

The following query returns GAUS.
:SENSe:BANDwidth:SHAPe?
:SENSe:BWIDth:SHAPe GAUSsian

[:SENSe]:BANDwidth|BWIDth:VIDeo

Syntax
[:SENSe]:BANDwidth|BWIDth:VIDeo <freq>
[:SENSe]:BANDwidth|BWIDth:VIDeo?

Description
Sets the video bandwidth (VBW).
Queries the video bandwidth.

Parameter

Name	Type	Range	Default
<freq>	Discrete	1 Hz to 10 MHz (at 1-3-10 step)	3 MHz

Remarks
This command is only available for GPSA mode.

Return Format
The query returns the video bandwidth in scientific notation. Its unit is Hz.

Example
The following command sets the video bandwidth to 1000 Hz.
:SENSe:BANDwidth:VIDeo 1000
:SENSe:BWIDth:VIDeo 1000

The following query returns 1.000000000e+03.
:SENSe:BANDwidth:VIDeo?
:SENSe:BWIDth:VIDeo?

[:SENSe]:BANDwidth|BWIDth:VIDeo:AUTO

Syntax
[:SENSe]:BANDwidth|BWIDth:VIDeo:AUTO <bool>
[:SENSe]:BANDwidth|BWIDth:VIDeo:AUTO?

Description
Enables or disables the auto setting mode of VBW.
Queries the status of the auto setting mode of VBW.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
This command is only available for GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the auto setting mode of VBW.
:SENSe:BANDwidth:VIDeo:AUTO ON or :SENSe:BANDwidth:VIDeo:AUTO 1
:SENSe:BWIDth:VIDeo:AUTO ON or :SENSe:BWIDth:VIDeo:AUTO 1

The following query returns 1.
:SENSe:BANDwidth:VIDeo:AUTO?
:SENSe:BWIDth:VIDeo:AUTO?

[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio

Syntax
[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio <number>
[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio?

Description
Sets the ratio of VBW to RBW.
Queries the V/R ratio.

Parameter

Name	Type	Range	Default
<number>	Discrete	0.00001 to 3000000 (at 1-3-10 step)	1

Remarks
This command is valid for the swept SA measurement in GPSA mode.

Return Format
The query returns the V/R ratio in scientific notation.

Example
The following command set the V/R ratio to 0.01.
:SENSe:BANDwidth:VIDeo:RATio 0.01
:SENSe:BWIDth:VIDeo:RATio 0.01

The following query returns 1.000000000e-02.
:SENSe:BANDwidth:VIDeo:RATio?
:SENSe:BWIDth:VIDeo:RATio?

[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio:AUTO

Syntax
[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio:AUTO <bool>
[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio:AUTO?

Description
Sets the VBW/RBW ratio auto state.
Queries the auto state of VBW/RBW ratio.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
This command is valid for the swept SA measurement in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the auto setting mode of V/R ratio.
:SENSe:BANDwidth:VIDeo:AUTO ON or :SENSe:BANDwidth:VIDeo:AUTO 1
:SENSe:BWIDth:VIDeo:AUTO ON or :SENSe:BWIDth:VIDeo:AUTO 1

The following query returns 1.
:SENSe:BANDwidth:VIDeo:AUTO?
:SENSe:BWIDth:VIDeo:AUTO?

[:SENSe]:BANDwidth:COUPle:AUTO

Syntax
[:SENSe]:BANDwidth:COUPle:AUTO

Description
Performs auto coupling for the band function.

[:SENSe]:CNRatio:OFFSet

Syntax
[:SENSe]:CNRatio:OFFSet <freq>
[:SENSe]:CNRatio:OFFSet?

Description
Sets the offset frequency. That is the difference between the center frequency of the carrier and that of the noise.
Queries the offset frequency.

Parameter

Name	Type	Range	Default
<freq>	Real	3 Hz to 2.166666 GHz	2 MHz

Remarks
This command is only valid when the C/N ratio measurement is enabled in GPSA mode.

Return Format
The query returns the offset frequency in integer. The unit is Hz.

Example
The following command sets the offset frequency to 1 MHz.
:SENSe:CNRatio:OFFSet 1000000

The following query returns 1000000.
:SENSe:CNRatio:OFFSet?

[:SENSe]:CNRatio:AVERage:COUNt

Syntax
[:SENSe]:CNRatio:AVERage:COUNt <integer>
[:SENSe]:CNRatio:AVERage:COUNt?

Description
Sets the average count of the C/N Ratio measurement.
Queries the average count of the C/N Ratio measurement.

Parameter

Name	Type	Range	Default
<integer>	Integer	1 to 1000	10

Remarks
This command is only valid when the C/N ratio measurement is enabled in GPSA mode.

Return Format
The query returns the average count in integer.

Example
The following command sets the average count to 100.
:SENSe:CNRatio:AVERage:COUNt 100

The following query returns 100.
:SENSe:CNRatio:AVERage:COUNt?

[:SENSe]:CNRatio:AVERage[:STATe]

Syntax
[:SENSe]:CNRatio:AVERage[:STATe] <bool>
[:SENSe]:CNRatio:AVERage[:STATe]?

Description
Sets the average state of C/N Ratio measurement.
Queries the average state of C/N Ratio measurement.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only valid when the C/N ratio measurement is enabled in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the average measurement function.
:SENSe:CNRatio:AVERage:STATe 1 or :SENSe:CNRatio:AVERage:STATe ON

The following query returns 1.
:SENSe:CNRatio:AVERage:STATe?

[:SENSe]:CNRatio:AVERage:TCONtrol

Syntax
[:SENSe]:CNRatio:AVERage:TCONtrol <enum>
[:SENSe]:CNRatio:AVERage:TCONtrol?

Description
Sets the average mode of the C/N Ratio measurement.
Queries the average mode of the C/N Ratio measurement.

Parameter

Name	Type	Range	Default
<enum>	Discrete	EXPonential|REPeat	EXPonential

Remarks
EXPonential: indicates the exponential average mode.
REPeat: indicates the repeat average mode.
When "EXPonential" is selected, the result is the exponential average of the measurement results obtained in the past N times (N is specified in *[:SENSe]:CNRatio:AVERage:COUNt*).
When "REPeat" is selected, the result is the arithmetic average of the measurement results obtained in the past N times (N is specified in *[:SENSe]:CNRatio:AVERage:COUNt*).
This command is only valid when the C/N ratio measurement is enabled in GPSA mode.

Return Format
The query returns EXP or REP.

Example
The following command sets the average mode to Repeat.
:SENSe:CNRatio:AVERage:TCONtrol REPeat

The following query returns REP.
:SENSe:CNRatio:AVERage:TCONtrol?

[:SENSe]:CNRatio:BANDwidth:INTegration

Syntax
[:SENSe]:CNRatio:BANDwidth:INTegration <freq>
[:SENSe]:CNRatio:BANDwidth:INTegration?

Description
Sets the carrier bandwidth.
Queries the carrier bandwidth.

Parameter

Name	Type	Range	Default
<freq>	Real	3 Hz to 2.166666 GHz	2 MHz

Remarks
This command is only valid when the C/N ratio measurement is enabled in GPSA mode.
The carrier bandwidth is related to the noise bandwidth. The range of the carrier bandwidth is from (noise bandwidth/20) to (noise bandwidth x 20).

Return Format
The query returns the carrier bandwidth in scientific notation. Its unit is Hz.

Example
The following command sets the carrier bandwidth to 1 MHz.
:SENSe:CNRatio:BANDwidth:INTegration 1000000 or :SENSe:CNRatio:BANDwidth:INTegration 1MHz

The following query returns 1.000000000e+06.
:SENSe:CNRatio:BANDwidth:INTegration?

[:SENSe]:CNRatio:BANDwidth:NOISe

Syntax
[:SENSe]:CNRatio:BANDwidth:NOISe <freq>
[:SENSe]:CNRatio:BANDwidth:NOISe?

Description
Sets the noise bandwidth.
Queries the noise bandwidth.

Parameter

Name	Type	Range	Default
<freq>	Real	3 Hz to 2.166666 GHz	2 MHz

Remarks
This command is only valid when the C/N ratio measurement is enabled in GPSA mode.

Return Format
The query returns the noise bandwidth in integer. Its unit is Hz.

Example
The following command sets the noise bandwidth to 1 MHz.
:SENSe:CNRatio:BANDwidth:NOISe 1000000

The following query returns 1000000.
:SENSe:CNRatio:BANDwidth:NOISe?

[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude]

Syntax
[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude] <enum>
[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude]?

Description
Sets the input impedance. Its unit is Ω.
Queries the input impedance.

Parameter

Name	Type	Range	Default
<enum>	Discrete	50|75	50

Remarks
If the output impedance of the system under test is 75 Ω, you should use a 75 Ω to 50 Ω adapter (option) supplied by RIGOL to connect the analyzer with the system under test, and then set the input impedance to 75 Ω.

Return Format
The query returns 50 or 75.

Example
The following command sets the input impedance to 75 Ω.
:SENSe:CORRection:IMPedance:INPut:MAGNitude 75

The following query returns 75.
:SENSe:CORRection:IMPedance:INPut:MAGNitude?

[:SENSe]:CORRection:SA[:RF]:GAIN

Syntax
[:SENSe]:CORRection:SA[:RF]:GAIN <rel_ampl>
[:SENSe]:CORRection:SA[:RF]:GAIN?

Description
Sets the external gain.
Queries the external gain.

Parameter

Name	Type	Range	Default
<rel_ampl>	Real	-120 dB to 120 dB	0 dB

Return Format
The query returns the external gain value in scientific notation. The unit is dB.

Example
The following command sets the external gain value to 20 dB.
:SENSe:CORRection:SA:RF:GAIN 20

The following query returns 2.000000000e+01.
:SENSe:CORRection:SA:RF:GAIN?

[:SENSe]:CORRection:CSET<n>[:STATe]

Syntax
[:SENSe]:CORRection:CSET<n>[:STATe] <bool>
[:SENSe]:CORRection:CSET<n>[:STATe]?

Description
Enables or disables the correction.
Queries the on/off status of the correction function.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4	-
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
The parameter <n> sets the amplitude correction type. 1 indicates Antenna, 2 indicates Cable, 3 indicates Other, and 4 indicates User.

Return Format
The query returns 0 or 1.

Example
The following command sets the amplitude correction type to cable.
:SENSe:CORRection:CSET1:STATe 1 or :SENSe:CORRection:CSET1:STATe ON

The following query returns 1.
:SENSe:CORRection:CSET1:STATe?

[:SENSe]:CORRection:CSET<n>:DESCription

Syntax
[:SENSe]:CORRection:CSET<n>:DESCription <string>
[:SENSe]:CORRection:CSET<n>:DESCription?

Description
Sets the description of the amplitude correction.
Queries the description of the amplitude correction.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4	-
<string>	ASCII String	The label can contain English letters and numbers, as well as some symbols.	-

Remarks
The parameter <n> sets the amplitude correction type. 1 indicates Antenna, 2 indicates Cable, 3 indicates Other, and 4 indicates User.

Return Format
The query returns the amplitude correction description in ASCII strings.

Example
The following command sets the amplitude correction description to Rigol.
:SENSe:CORRection:CSET1:DESCription Rigol

The following query returns Rigol.
:SENSe:CORRection:CSET1:DESCription?

[:SENSe]:CORRection:CSET<n>:COMMent

Syntax
[:SENSe]:CORRection:CSET<n>:COMMent <string>
[:SENSe]:CORRection:CSET<n>:COMMent?

Description
Sets the comments of the amplitude correction.
Queries the comments of the amplitude correction.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4	-
<string>	ASCII String	The label can contain English letters and numbers, as well as some symbols.	-

Remarks
The parameter <n> sets the amplitude correction type. 1 indicates Antenna, 2 indicates Cable, 3 indicates Other, and 4 indicates User.

Return Format
The query returns the comments of the amplitude correction in ASCII strings.

Example
The following command sets the amplitude correction comments to Rigol.
:SENSe:CORRection:CSET1:COMMent Rigol

The following query returns Rigol.
:SENSe:CORRection:CSET1:COMMent?

[:SENSe]:CORRection:CSET<n>:X:SPACing

Syntax
[:SENSe]:CORRection:CSET<n>:X:SPACing <enum>
[:SENSe]:CORRection:CSET<n>:X:SPACing?

Description
Sets the frequency interpolation type.
Queries the frequency interpolation type.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4	-
<enum>	Discrete	LINear|LOG	LINear

Remarks
The parameter <n> sets the amplitude correction type. 1 indicates Antenna, 2 indicates Cable, 3 indicates Other, and 4 indicates User.
LINear: linear type.
LOG: log type.

Return Format
The query returns LIN or LOG.

Example
The following command sets the amplitude interpolation type of the antenna amplitude correction to LOG.
:SENSe:CORRection:CSET1:X:SPACing LOG

The following query returns LOG.
:SENSe:CORRection:CSET1:X:SPACing?

[:SENSe]:CORRection:CSET<n>:DATA

Syntax
[:SENSe]:CORRection:CSET<n>:DATA <string>
[:SENSe]:CORRection:CSET<n>:DATA?

Description
Sets the list table of the amplitude correction.
Queries the list table of the amplitude correction.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4	-
<string>	ASCII String	The label can contain English letters and numbers, as well as some symbols.	-

Remarks
The parameter <n> sets the amplitude correction type. 1 indicates Antenna, 2 indicates Cable, 3 indicates Other, and 4 indicates User.

Return Format
The query returns the list table of the amplitude correction in ASCII strings.

Example
The following command sets the list table amplitude correction comments to Rigol.
:SENSe:CORRection:CSET1:DATA

The following query returns Rigol.
:SENSe:CORRection:CSET1:DATA?

[:SENSe]:CORRection:CSET<n>:DATA:MERG

Syntax
[:SENSe]:CORRection:CSET<n>:DATA:MERG <string>

Description
Merges the amplitude correction.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4	-
<string>	ASCII String	The label can contain English letters and numbers, as well as some symbols.	-

Remarks
The parameter <n> sets the amplitude correction type. 1 indicates Antenna, 2 indicates Cable, 3 indicates Other, and 4 indicates User.

[:SENSe]:CORRection:CSET<n>:DELete

Syntax
[:SENSe]:CORRection:CSET<n>:DELete

Description
Deletes the single amplitude correction data.

Remarks
The parameter <n> sets the amplitude correction type. 1 indicates Antenna, 2 indicates Cable, 3 indicates Other, and 4 indicates User.

[:SENSe]:CORRection:CSET:ALL:DELete

Syntax
[:SENSe]:CORRection:CSET:ALL:DELete

Description
Delete all amplitude correction data.

Remarks
The parameter <n> sets the amplitude correction type. 1 indicates Antenna, 2 indicates Cable, 3 indicates Other, and 4 indicates User.

[:SENSe]:CORRection:CSET:ALL[:STATe]

Syntax
[:SENSe]:CORRection:CSET:ALL[:STATe] <bool>
[:SENSe]:CORRection:CSET:ALL[:STATe]?

Description
Enables or disables the amplitude correction.
Queries whether to enable the amplitude correction.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4	-
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
The parameter <n> sets the amplitude correction type. 1 indicates Antenna, 2 indicates Cable, 3 indicates Other, and 4 indicates User.

Return Format
The query returns 0 or 1.

Example
The following command sets the amplitude correction type to cable.
:SENSe:CORRection:CSET:STATe 1 or :SENSe:CORRection:CSET:STATe ON

The following query returns 1.
:SENSe:CORRection:CSET:STATe?

[:SENSe]:CORRection:CSET[:ID]

Syntax
x[:SENSe]:CORRection:CSET>:ID <enum>
[:SENSe]:CORRection:CSET[:ID]?

Description
Sets the type of the amplitude correction.
Queries the comments of the amplitude correction.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4	-
<enum>	Discrete	OFF|ANTEnna|CABLe|OTHEr|USER	-

Remarks
ANTEnna: Antenna.
CABLe: Cable.
OTHEr: Other.
USER: indicates user-defined.

Return Format
The query returns ANTE, CABL, OTHE, or USER.

Example
The following command sets the amplitude correction type to OTHEr.
:SENSe:CORRection:CSET:ID OTHEr

The following query returns OTHE.
:SENSe:CORRection:CSET:ID?

[:SENSe]:DETector:TRACe:PVTime

Syntax
[:SENSe]:DETector:TRACe:PVTime <type>
[:SENSe]:DETector:TRACe:PVTime?

Description
Sets the detector type of the trace in the PvT view.
Queries the detector type for the trace in the PvT view.

Parameter

Name	Type	Range	Default
<type>	Discrete	AVERage|NEGative|POSitive|SAMPle	POSitive

Remarks
AVERage: indicates the voltage average.
NEGative: indicates the negative peak.
POSitive: indicates the positive peak.
SAMPle: indicates the sample detector.
This command is only available for RTSA mode.

Return Format
The query returns AVER, NEG, POS, or SAMP.

Example
The following command sets the detector type of the trace to POSitive.
:SENSe:DETector:TRACe:PVTime POSitive

The following query returns POS.
:SENSe:DETector:TRACe:PVTime?

[:SENSe]:DETector:TRACe:PVTime:AUTO

Syntax
[:SENSe]:DETector:TRACe:PVTime:AUTO <bool>
[:SENSe]:DETector:TRACe:PVTime:AUTO?

Description
Enables or disables the PvT detector auto.
Queries the PvT detector auto.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
N/A

Return Format
The query returns 0 or 1.

Example
The following command enables or disables the PvT detector auto.
:SENSe:DETector:TRACe:PVTime:AUTO ON or :SENSe:DETector:TRACe:PVTime:AUTO 1

The following query returns 1.
:SENSe:DETector:TRACe:PVTime:AUTO?

[:SENSe]:DETector:TRACe<n><type>

Syntax
[:SENSe]:DETector:TRACe<n> <type>
[:SENSe]:DETector:TRACe<n>?

Description
Sets the detector type for the specified trace.
Queries the detector type for the specified trace.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1
<type>	Discrete	AVERage（VAVerage）|NEGative|NORMal|POSitive|SAMPle|QPEak|RAVerage（RMS）	POSitive

Remarks
AVERage|VAVerage: indicates the voltage average.
NEGative: indicates the negative peak.
NORMal: indicates the normal detector.
POSitive: indicates the positive peak.
SAMPle: indicates the sample detector.
QPEak: indicates the quasi-peak.
RAVerage|RMS: indicates the RMS average.
In RTSA mode, the parameters in this command only include AVERage, NEGative, POSitive, and SAMPle.

Return Format
The query returns AVER, NEG, NORM, POS, SAMP, QPE, or RAV.

Example
The following command sets the detector type of Trace 1 to Positive.
:SENSe:DETector:TRACe1 POSitive

The following query returns POS.
:SENSe:DETector:TRACe1?

[:SENSe]:DETector:TRACe<n>:AUTO

Syntax
[:SENSe]:DETector:TRACe<n>:AUTO <bool>
[:SENSe]:DETector:TRACe<n>:AUTO?

Description
Enables or disables the Detector Auto function for the specified trace.
Queries the status of the Detector Auto function for the specified trace.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	1
<bool>	Bool	OFF|ON|0|1	ON|1

Return Format
The query returns 0 or 1.

Example
The following command enables the Detector Auto function for Trace 1.
:SENSe:DETector:TRACe1:AUTO ON or :SENSe:DETector:TRACe1:AUTO 1

The following query returns 1.
:SENSe:DETector:TRACe1:AUTO?

[:SENSe]:FREQuency:CENTer

Syntax
[:SENSe]:FREQuency:CENTer <freq>
[:SENSe]:FREQuency:CENTer?

Description
Sets the center frequency.
Queries the center frequency.

Parameter

Name	Type	Range	Default
<freq>	Real	(Smin[2]/2) to (Fmax - Smin/2)	Fmax[1]/2

Note[1]: Fmax is determined by the instrument model. Refer to the product model list in the Overview chapter of the document.
Note[2]: Smin indicates the minimum span in non-zero mode.

Return Format
The query returns the center frequency in scientific notation. The unit is Hz.

Example
The following command sets the center frequency to 1 MHz.
:SENSe:FREQuency:CENTer 1000000

The following query returns 1.000000000e+06.
:SENSe:FREQuency:CENTer?

[:SENSe]:FREQuency:CENTer:STEP:AUTO

Syntax
[:SENSe]:FREQuency:CENTer:STEP:AUTO <bool>
[:SENSe]:FREQuency:CENTer:STEP:AUTO?

Description
Enables or disables the auto setting mode of the CF step.
Queries the status of the auto setting mode of the CF step.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Return Format
The query returns 0 or 1.

Example
The following command enables the auto setting mode of the CF step.
:SENSe:FREQuency:CENTer:STEP:AUTO ON or :SENSe:FREQuency:CENTer:STEP:AUTO 1

The following query returns 1.
:SENSe:FREQuency:CENTer:STEP:AUTO?

[:SENSe]:FREQuency:CENTer:STEP[:INCRement]

Syntax
[:SENSe]:FREQuency:CENTer:STEP[:INCRement] <freq>
[:SENSe]:FREQuency:CENTer:STEP[:INCRement]?

Description
Sets the CF step.
Queries the CF step.

Parameter

Name	Type	Range	Default
<freq>	Real	-Fmax to Fmax	Fmax/10

Return Format
The query returns the center frequency step in scientific notation. The unit is Hz.

Example
The following command sets the CF step to 100 kHz.
:SENSe:FREQuency:CENTer:STEP:INCRement 100000

The following query returns 1.000000000e+05.
:SENSe:FREQuency:CENTer:STEP:INCRement?

[:SENSe]:FREQuency:OFFSet

Syntax
[:SENSe]:FREQuency:OFFSet <freq>
[:SENSe]:FREQuency:OFFSet?

Description
Sets the frequency offset.
Queries the frequency offset.

Parameter

Name	Type	Range	Default
<freq>	Real	-500 GHz to 500 GHz	0 Hz

Remarks
The change of this parameter only changes the display values of the center frequency, start frequency, and stop frequency; but does not affect any hardware settings of the spectrum analyzer.

Return Format
The query returns the frequency offset in scientific notation. The unit is Hz.

Example
The following command sets the frequency offset to 1 MHz.
:SENSe:FREQuency:OFFSet 1000000

The following query returns 1.000000000e+06.
:SENSe:FREQuency:OFFSet?

[:SENSe]:FREQuency:SPAN

Syntax
[:SENSe]:FREQuency:SPAN <freq>
[:SENSe]:FREQuency:SPAN?

Description
Sets the span.
Queries the span.

Parameter

Name	Type	Range	Default
<freq>	Real	GPSA mode: 0 Hz, 10 Hz to F_{max} RTSA mode: 5 kHz to 20 MHz (CF ≤ 100 MHz) 5 kHz to 40 MHz (CF > 100 MHz)	GPSA mode: F_{max} RTSA mode: 40 MHz

Remarks
The span can be set to 0 only in GPSA mode. After the span is set to 0, the instrument enters the zero span mode; the X axis changes from frequency to time, and the instrument only displays signals whose frequency equals to the center frequency.

Return Format
The query returns the span in scientific notation. The unit is Hz.

Example
The following command sets the span to 20 MHz.
:SENSe:FREQuency:SPAN 20000000

The following query returns 2.000000000e+07.
:SENSe:FREQuency:SPAN?

[:SENSe]:FREQuency:SPAN:BANDwidth[:RESolution]:RATio

Syntax
[:SENSe]:FREQuency:SPAN:BANDwidth[:RESolution]:RATio <integer>
[:SENSe]:FREQuency:SPAN:BANDwidth[:RESolution]:RATio?

Description
Sets the ratio of span to RBW.
Queries the ratio of span to RBW.

Parameter

Name	Type	Range	Default
<integer>	Discrete	2 to 10000	106

Remarks
This command is valid for the swept SA measurement in GPSA mode.

Return Format
The query returns the span/RBW ratio in integer.

Example
The following command sets the span/RBW ratio to 100.
:SENSe:FREQuency:SPAN:BANDwidth:RESolution:RATio 100

The following query returns 100.
:SENSe:FREQuency:SPAN:BANDwidth:RESolution:RATio?

[:SENSe]:FREQuency:SPAN:BANDwidth[:RESolution]:RATio:AUTO

Syntax
[:SENSe]:FREQuency:SPAN:BANDwidth[:RESolution]:RATio:AUTO <bool>
[:SENSe]:FREQuency:SPAN:BANDwidth[:RESolution]:RATio:AUTO?

Description
Enables or disables the auto setting mode of span/bandwidth ratio.
Queries the status of the auto setting mode of span/bandwidth ratio.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
This command is valid for the swept SA measurement in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the auto setting mode of span/bandwidth ratio.
:SENSe:FREQuency:SPAN:BANDwidth:RESolution:RATio:AUTO ON
or :SENSe:FREQuency:SPAN:BANDwidth:RESolution:RATio:AUTO 1

The following query returns 1.
:SENSe:FREQuency:SPAN:BANDwidth:RESolution:RATio:AUTO?

[:SENSe]:FREQuency:SPAN:FULL

Syntax
[:SENSe]:FREQuency:SPAN:FULL

Description
Sets the maximum span (full span).

[:SENSe]:FREQuency:SPAN:PREVious

Syntax
[:SENSe]:FREQuency:SPAN:PREVious

Description
Sets the span to the last set value.

[:SENSe]:FREQuency:SPAN:ZERO

Syntax
[:SENSe]:FREQuency:SPAN:ZERO

Description
Sets the span to 0 (zero span).

Remarks
This command is only valid in GPSA mode.

[:SENSe]:FREQuency:PVTime:SPAN

Syntax
[:SENSe]:FREQuency:PVTime:SPAN <freq>
[:SENSe]:FREQuency:PVTime:SPAN?

Description
Sets the PvT span.
Queries the PVT span.

Parameter

Name	Type	Range	Default
<freq>	Real	5 kHz to 20 MHz (CF ≤ 100 MHz) 5 kHz to 40 MHz (CF > 100 MHz)	40 MHz

Remarks
This command is only valid in RTSA mode.

Return Format
The query returns the span in scientific notation. The unit is Hz.

Example
The following command sets the span to 20 MHz.
:SENSe:FREQuency:SPAN 20000000

The following query returns 2.000000000e+07.
:SENSe:FREQuency:SPAN?

[:SENSe]:FREQuency:STARt

Syntax
[:SENSe]:FREQuency:STARt <freq>
[:SENSe]:FREQuency:STARt?

Description
Sets the start frequency.
Queries the start frequency.

Parameter

Name	Type	Range[1]	Default
<freq>	Real	0 Hz to Fmax	center frequency - span/2

Note[1]: The range is from 0 Hz to (Fmax - 10 Hz) in non-zero span.

Return Format
The query returns the start frequency in scientific notation. The unit is Hz.

Example
The following command sets the start frequency to 100 MHz.
:SENSe:FREQuency:STARt 100000000

The following query returns 1.000000000e+08.
:SENSe:FREQuency:STARt?

[:SENSe]:FREQuency:STOP

Syntax
[:SENSe]:FREQuency:STOP <freq>
[:SENSe]:FREQuency:STOP?

Description
Sets the stop frequency.
Queries the stop frequency.

Parameter

Name	Type	Range[1]	Default
<freq>	Real	0 Hz to Fmax	center frequency + span/2

Note[1]: The range is from 10 Hz to Fmax in non-zero span.

Return Format
The query returns the stop frequency in scientific notation. The unit is Hz.

Example
The following command sets the stop frequency to 10 MHz.
:SENSe:FREQuency:STOP 10000000

The following query returns 1.000000000e+07.
:SENSe:FREQuency:STOP?

[:SENSe]:FREQuency:TUNE:IMMediate

Syntax
[:SENSe]:FREQuency:TUNE:IMMediate

Description
Automatically searches for the signal within the full frequency range, and adjusts the frequency and amplitude for optimal display effect of the signal.

Remarks
This command is invalid when the advanced measurement is enabled in GPSA mode or when in RTSA mode.

[:SENSe]:OBWidth:AVERage:COUNt

Syntax
[:SENSe]:OBWidth:AVERage:COUNt <integer>
[:SENSe]:OBWidth:AVERage:COUNt?

Description
Sets the average count of the occupied bandwidth measurement.
Queries the average count of the occupied bandwidth measurement.

Parameter

Name	Type	Range	Default
<integer>	Integer	1 to 1000	10

Remarks
This command is only valid when the occupied bandwidth measurement is enabled in GPSA mode.

Return Format
The query returns the average count in integer.

Example
The following command sets the average count to 100.
:SENSe:OBWidth:AVERage:COUNt 100

The following query returns 100.
:SENSe:OBWidth:AVERage:COUNt?

[:SENSe]:OBWidth:AVERage[:STATe]

Syntax
[:SENSe]:OBWidth:AVERage[:STATe] <bool>
[:SENSe]:OBWidth:AVERage[:STATe]?

Description
Enables or disables the average measurement function of the occupied bandwidth measurement.
Queries the status of the average measurement function of the occupied bandwidth measurement.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
This command is only valid when the occupied bandwidth measurement is enabled in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the average measurement function.
:SENSe:OBWidth:AVERage:STATe ON or :SENSe:ACPower:AVERage:STATe 1

The following query returns 1.
:SENSe:OBWidth:AVERage:STATe?

[:SENSe]:OBWidth:AVERage:TCONtrol

Syntax
[:SENSe]:OBWidth:AVERage:TCONtrol <type>
[:SENSe]:OBWidth:AVERage:TCONtrol?

Description
Selects the average mode of the occupied bandwidth measurement.
Queries the average mode of the occupied bandwidth measurement.

Parameter

Name	Type	Range	Default
<type>	Discrete	EXPonential|REPeat	EXPonential

Remarks
EXPonential: indicates the exponential average.
REPeat: indicates the repeat average.
When "EXPonential" is selected, the result is the exponential average of the measurement results obtained in the past N times (N is specified in[:SENSe]:OBWidth:AVERage:COUNt).
When "REPeat" is selected, the result is the arithmetic average of the measurement results obtained in the past N times (N is determines in[:SENSe]:OBWidth:AVERage:COUNt).
This command is only valid when the occupied bandwidth measurement is enabled in GPSA mode.

Return Format
The query returns EXP or REP.

Example
The following command sets the average mode to Repeat.
:SENSe:OBWidth:AVERage:TCONtrol REPeat

The following query returns REP.
:SENSe:OBWidth:AVERage:TCONtrol?

[:SENSe]:OBWidth:XDB

Syntax
[:SENSe]:OBWidth:XDB <integer>
[:SENSe]:OBWidth:XDB?

Description
Sets the EBW X dB.
Queries the EBW X dB.

Parameter

Name	Type	Range	Default
<integer>	Real	-100 dB to -0.1 dB	-10 dB

Remarks
This command is only valid when the occupied bandwidth measurement is enabled in GPSA mode.

Return Format
The query returns the EBW X dB value in scientific notation.

Example
The following command sets the EBW X dB value to -15 dB.
:SENSe:OBWidth:XDB -15 dB

The following query returns -1.5E+01.
:SENSe:OBWidth:XDB?

[:SENSe]:OBWidth:MAXHold

Syntax
[:SENSe]:OBWidth:MAXHold <bool>
[:SENSe]:OBWidth:MAXHold?

Description
Enables or disables Max Hold of the occupied bandwidth measurement.
Queries the Max Hold state of the occupied bandwidth measurement.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only valid when the occupied bandwidth measurement is enabled in GPSA mode.
When Max Hold is enabled, each measurement result is compared with the previous result, and then display whichever is the maximum.
When Max Hold is disabled, the current measurement result is displayed.
Max Hold and average measurement mode are mutually exclusive. When Max Hold is enabled, the average measurement mode will be automatically disabled.

Return Format
The query returns 0 or 1.

Example
The following command enables the Max Hold.
:SENSe:OBWidth:MAXHold ON or :SENSe:OBWidth:MAXHold 1

The following query returns 1.
:SENSe:OBWidth:MAXHold?

[:SENSe]:OBWidth:PERCent

Syntax
[:SENSe]:OBWidth:PERCent <real>
[:SENSe]:OBWidth:PERCent?

Description
Sets the percentage the signal power takes up in the whole span power (power ratio).
Queries the power ratio of the occupied bandwidth measurement.

Parameter

Name	Type	Range	Default
<real>	Real	1 to 99.99	99

Remarks
This command is only valid when the occupied bandwidth measurement is enabled in GPSA mode.
The range of <real> is from 1 to 99.99. Therefore, set the range of the corresponding parameter for the analyzer to "1%-99.99%".

Return Format
The query returns the percentage in scientific notation.

Example

The following command sets the power ratio to 90%.
:SENSe:OBWidth:PERCent 90
The following query returns 9.000000000e+01.
:SENSe:OBWidth:PERCent?

[:SENSe]:CHPower:DELete

Syntax
[:SENSe]:CHPower:DELete

Description
Deletes all channels.

[:SENSe]:CHPower:CHANnel

Syntax
[:SENSe]:CHPower:CHANnel <string>
[:SENSe]:CHPower:CHANnel?

Description
Sets the channel parameters.
Queries the channel parameters.

Parameter

Name	Type	Range	Default
<string>	ASCII String	——	——

Remarks
This command is only valid when the CH power measurement is enabled in GPSA mode.

Return Format
The query returns the channel parameters in strings.

Example
The following command sets the channel parameter to 8.25 GHz 13.5 GHz.
:SENSe:CHPower:CHANnel 8.25,13.5

The following query returns 8.250000,1300000.
:SENSe:CHPower:CHANnel?

[:SENSe]:CHPower:AVERage:COUNt

Syntax
[:SENSe]:CHPower:AVERage:COUNt <integer>
[:SENSe]:CHPower:AVERage:COUNt?

Description
Sets the average count of the multi-channel power measurement.
Queries the average count of the the multi-channel power measurement.

Parameter

Name	Type	Range	Default
<integer>	Integer	1 to 1000	10

Remarks

This command is only valid when the multi-channel power measurement is enabled in GPSA mode.

Return Format
The query returns the average count in integer.

Example
The following command sets the average count to 100.
:SENSe:CHPower:AVERage:COUNt 100

The following query returns 100.
:SENSe:CHPower:AVERage:COUNt?

[:SENSe]:CHPower:AVERage[:STATe]

Syntax
[:SENSe]:CHPower:AVERage[:STATe] <bool>
[:SENSe]:CHPower:AVERage[:STATe]?

Description
Enables or disables the average state of the channel power measurement.
Queries the average state of the channel power measurement.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
This command is only valid when the multi-channel power measurement is enabled in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the average measurement function.
:SENSe:CHPower:AVERage:STATe ON or :SENSe:CHPower:AVERage:STATe 1

The following query returns 1.
:SENSe:CHPower:AVERage:STATe?

[:SENSe]:MCHPower:AVERage:TCONtrol

Syntax
[:SENSe]:CHPower:AVERage:TCONtrol <type>
[:SENSe]:CHPower:AVERage:TCONtrol?

Description
Selects the average mode of the multi-channel power measurement.
Queries the average mode of the multi-channel power measurement.

Parameter

Name	Type	Range	Default
<type>	Discrete	EXPonential|REPeat	EXPonential

Remarks
EXPonential: indicates the exponential average.
REPeat: indicates the repeat average.

When "EXPonential" is selected, the result is the exponential average of the measurement results obtained in the past N times (N is specified in *[:SENSe]:CHPower:AVERage:COUNt*).
When "REPeat" is selected, the result is the arithmetic average of the measurement results obtained in the past N times (N is specified in *[:SENSe]:CHPower:AVERage:COUNt*).
This command is only valid when the channel power measurement is enabled in GPSA mode.

Return Format
The query returns EXP or REP.

Example
The following command sets the average mode to Repeat.
:SENSe:CHPower:AVERage:TCONtrol REPeat

The following query returns REP.
:SENSe:CHPower:AVERage:TCONtrol?

[:SENSe]:POWer[:RF]:ATTenuation

Syntax
[:SENSe]:POWer[:RF]:ATTenuation <real>
[:SENSe]:POWer[:RF]:ATTenuation?

Description
Sets the attenuation of the RF front-end attenuator.
Queries the attenuation of the RF front-end attenuator.

Parameter

Name	Type	Range	Default
<real>	Integer	0 dB to 50 dB	10 dB

Return Format
The query returns the attenuation in integer. The unit is dB.

Example
The following command sets the attenuation to 20 dB.
:SENSe:POWer:RF:ATTenuation 20

The following query returns 20.
:SENSe:POWer:RF:ATTenuation?

[:SENSe]:POWer[:RF]:ATTenuation:AUTO

Syntax
[:SENSe]:POWer[:RF]:ATTenuation:AUTO <bool>
[:SENSe]:POWer[:RF]:ATTenuation:AUTO?

Description
Enables or disables the auto setting mode of the input attenuation.
Queries the status of the auto setting mode of the input attenuation.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Return Format
The query returns 0 or 1.

Example
The following command disables the auto setting mode of the input attenuation.
:SENSe:POWer:RF:ATTenuation:AUTO OFF or :SENSe:POWer:RF:ATTenuation:AUTO 0

The following query returns 0.
:SENSe:POWer:RF:ATTenuation:AUTO?

[:SENSe]:POWer[:RF]:GAIN[:STATe]

Syntax
[:SENSe]:POWer[:RF]:GAIN[:STATe] <bool>
[:SENSe]:POWer[:RF]:GAIN[:STATe]?

Description
Enables or disables the preamplifier.
Queries the on/off status of the preamplifier.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 0 or 1.

Example
The following command enables the preamplifier.
:SENSe:POWer:RF:GAIN:STATe ON or :SENSe:POWer:RF:GAIN:STATe 1

The following query returns 1.
:SENSe:POWer:RF:GAIN:STATe?

[:SENSe]:POWer[:RF]:MIXer:RANGe[:UPPer]

Syntax
[:SENSe]:POWer[:RF]:MIXer:RANGe[:UPPer] <ampl>
[:SENSe]:POWer[:RF]:MIXer:RANGe[:UPPer]?

Description
Sets the maximum power of the input mixer.
Queries the maximum power of the input mixer.

Parameter

Name	Type	Range	Default
<ampl>	Real	-50 dBm to -10 dBm	-10 dBm

Return Format
The query returns the maximum power of the input mixer in scientific notation. The unit is dBm.

Example
The following command sets the maximum power of the input mixer to -20 dBm.
:SENSe:POWer:RF:MIXer:RANGe:UPPer -20

The following query returns -2.000000000e+01.
:SENSe:POWer:RF:MIXer:RANGe:UPPer?

[:SENSe]:COUPle:AUTO

Syntax
[:SENSe]:COUPle:AUTO

Description
Performs auto coupling.

[:SENSe]:SWEep:POINts

Syntax
[:SENSe]:SWEep:POINts <integer>
[:SENSe]:SWEep:POINts?

Description
Sets the number of sweep points.
Queries the number of sweep points.

Parameter

Name	Type	Range	Default
<integer>	Integer	101 to 10001	801

Remarks
This command is only valid in GPSA mode.

Return Format
The query returns the number of sweep points in integer.

Example
The following command sets the number of sweep points to 650.
:SENSe:SWEep:POINts 650

The following query returns 650.
:SENSe:SWEep:POINts?

[:SENSe]:SWEep:TIME

Syntax
[:SENSe]:SWEep:TIME <time>
[:SENSe]:SWEep:TIME?

Description
Sets the sweep time.
Queries the sweep time.

Parameter

Name	Type	Range	Default
<time>	Real	1 ms to 4,000 s (non-zero span) 1 us to 6,000 s (zero span)	1 ms

Remarks
This command is only valid in GPSA mode.

Return Format

The query returns the sweep time in scientific notation. The unit is s.

Example
The following command sets the sweep time to 100 ms.
:SENSe:SWEep:TIME 0.1

The following query returns 1.000000000e-01.
:SENSe:SWEep:TIME?

[:SENSe]:SWEep:TIME:AUTO

Syntax
[:SENSe]:SWEep:TIME:AUTO <bool>
[:SENSe]:SWEep:TIME:AUTO?

Description
Enables or disables auto sweep time.
Queries the status of the auto sweep time.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
This command is only valid in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the auto sweep time.
:SENSe:SWEep:TIME:AUTO ON or :SENSe:SWEep:TIME:AUTO 1

The following query returns 1.
:SENSe:SWEep:TIME:AUTO?

[:SENSe]:SWEep:TIME:AUTO:RULes

Syntax
[:SENSe]:SWEep:TIME:AUTO:RULes <enum>
[:SENSe]:SWEep:TIME:AUTO:RULes?

Description
Sets the sweep time rule.
Queries the sweep time rule.

Parameter

Name	Type	Range	Default
<enum>	Discrete	NORMal|ACCuracy	NORMal

Remarks
NORMal: The sweep will be faster if "Normal" is selected.
ACCuracy: a higher measurement precision can be obtained.
This command is only valid in GPSA mode.

Return Format

The query returns NORM or ACC.

Example
The following command sets the sweep time rule to Accuracy.
:SENSe:SWEep:TIME:AUTO:RULes ACCuracy

The following query returns ACC.
:SENSe:SWEep:TIME:AUTO:RULes?

[:SENSe]:SWEep:TYPE

Syntax
[:SENSe]:SWEep:TYPE <type>
[:SENSe]:SWEep:TYPE?

Description
Sets the sweep type.
Queries the sweep type.

Parameter

Name	Type	Range	Default
<type>	Discrete	SWEep|FFT	SWEep

Remarks
This command is only valid in GPSA mode.

Return Format
The query returns SWE or FFT.

Example
The following command sets the sweep type to FFT.
[:SENSe]:SWEep:TYPE FFT

The following query returns FFT.
[:SENSe]:SWEep:TYPE?

[:SENSe]:SWEep:TYPE:AUTO

Syntax
[:SENSe]:SWEep:TYPE:AUTO <bool>
[:SENSe]:SWEep:TYPE:AUTO?

Description
Sets the sweep type mode.
Queries the sweep type mode.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
This command is only valid in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the auto sweep type mode.
:SENSe:SWEep:TYPE:AUTO ON or :SENSe:SWEep:TYPE:AUTO 1

The following query returns 1.
:SENSe:SWEep:TYPE:AUTO?

[:SENSe]:TOI:FREQuency:BASE:LOWer

Syntax
[:SENSe]:TOI:FREQuency:BASE:LOWer <freq>
[:SENSe]:TOI:FREQuency:BASE:LOWer?

Description
Sets the Base Lower.
Queries the Base Lower.

Parameter

Name	Type	Range	Default
<freq>	Real	10 Hz to F_{max}	——

Remarks
This command is only valid when the TOI measurement is enabled in GPSA mode.

Return Format
The query returns the Base Lower in scientific notation.

Example
The following command sets Base Lower to 4.249 GHz.
:SENSe:TOI:FREQuency:BASE:LOWer 4.249 GHz

The following query returns 4.249E+09.
:SENSe:TOI:FREQuency:BASE:LOWer?

[:SENSe]:TOI:FREQuency:BASE:LOWer:AUTO

Syntax
[:SENSe]:TOI:FREQuency:BASE:LOWer:AUTO <bool>
[:SENSe]:TOI:FREQuency:BASE:LOWer:AUTO?

Description
Sets the lower base freq mode.
Queries the lower based freq mode.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
This command is only valid when the TOI measurement is enabled in GPSA mode.
ON|1: sets the lower base freq mode to Auto.
OFF|0: sets the lower base freq mode to Manual.

Return Format
The query returns 0 or 1.

Example
The following command sets the lower base freq mode to Auto.
:SENSe:TOI:FREQuency:BASE:LOWer:AUTO 1 or :SENSe:TOI:FREQuency:BASE:LOWer:AUTO ON

The following query returns 1.
:SENSe:TOI:FREQuency:BASE:LOWer:AUTO?

[:SENSe]:TOI:FREQuency:BASE:UPPer

Syntax
[:SENSe]:TOI:FREQuency:BASE:UPPer <freq>
[:SENSe]:TOI:FREQuency:BASE:UPPer?

Description
Sets the Base Upper.
Queries the Base Upper.

Parameter

Name	Type	Range	Default
<freq>	Real	11 Hz to F_{max}	——

Remarks
This command is only valid when the TOI measurement is enabled in GPSA mode.

Return Format
The query returns the Base Upper in scientific notation.

Example
The following command sets the Base Upper to 4.249 GHz.
:SENSe:TOI:FREQuency:BASE:UPPer 4.249 GHz

The following query returns 4.249E+09.
:SENSe:TOI:FREQuency:BASE:UPPer?

[:SENSe]:TOI:FREQuency:BASE:UPPer:AUTO

Syntax
[:SENSe]:TOI:FREQuency:BASE:UPPer:AUTO <bool>
[:SENSe]:TOI:FREQuency:BASE:UPPer:AUTO?

Description
Sets the upper base freq mode.
Queries the upper base freq mode.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
This command is only valid when the TOI measurement is enabled in GPSA mode.
ON|1: sets the upper base freq mode to Auto.
OFF|0: sets the upper base freq mode to Manual.

Return Format
The query returns 0 or 1.

Example
The following command sets the upper base freq mode to Auto.
:SENSe:TOI:FREQuency:BASE:UPPer:AUTO 1 or :SENSe:TOI:FREQuency:BASE:UPPer:AUTO ON

The following query returns 1.
:SENSe:TOI:FREQuency:BASE:UPPer:AUTO?

[:SENSe]:TOI:AVERage:COUNt

Syntax
[:SENSe]:TOI:AVERage:COUNt <integer>
[:SENSe]:TOI:AVERage:COUNt?

Description
Sets the average count of the TOI measurement.
Queries the average count of the TOI measurement.

Parameter

Name	Type	Range	Default
<integer>	Integer	1 to 1000	10

Remarks
This command is only valid when the TOI measurement is enabled in GPSA mode.

Return Format
The query returns the average count in integer.

Example
The following command sets the average count to 100.
:SENSe:TOI:AVERage:COUNt 100

The following query returns 100.
:SENSe:TOI:AVERage:COUNt?

[:SENSe]:TOI:AVERage[:STATe]

Syntax
[:SENSe]:TOI:AVERage[:STATe] <bool>
[:SENSe]:TOI:AVERage[:STATe]?

Description
Enables or disables the average state of the TOI measurement.
Queries the average state of the TOI measurement.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	ON|1

Remarks
This command is only valid when the TOI measurement is enabled in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the average measurement function.

:SENSe:TOI:AVERage:STATe ON or :SENSe:TOI:AVERage:STATe 1

The following query returns 1.
:SENSe:TOI:AVERage:STATe?

[:SENSe]:TOI:AVERage:TCONtrol

Syntax
[:SENSe]:TOI:AVERage:TCONtrol <enum>
[:SENSe]:TOI:AVERage:TCONtrol?

Description
Sets the average mode of the TOI measurement.
Queries the average mode of the TOI measurement.

Parameter

Name	Type	Range	Default
<enum>	Discrete	EXPonential|REPeat	EXPonential

Remarks
EXPonential: indicates the exponential average.
REPeat: indicates the repeat average.
When "EXPonential" is selected, the result is the exponential average of the measurement results obtained in the past N times.
When "REPeat" is selected, the result is the arithmetic average of the measurement results obtained in the past N times.
This command is only valid when the TOI measurement is enabled in GPSA mode.

Return Format
The query returns EXP or REP.

Example
The following command sets the average mode to Repeat.
:SENSe:TOI:AVERage:TCONtrol REPeat

The following query returns REP.
:SENSe:TOI:AVERage:TCONtrol?

[:SENSe]:TPOWer:SUMMary

Syntax
[:SENSe]:TPOWer:SUMMary<enum>
[:SENSe]:TPOWer:SUMMary?

Description
Sets the TP type.
Queries the TP type.

Parameter

Name	Type	Range	Default
<enum>	Discrete	PEAK|AVERage|RMS	PEAK

Remarks
This command is only valid when the time-domain power measurement is enabled in GPSA mode.
PEAK: indicates peak power.
AVERage: indicates average power.

RMS: indicates the RMS power.

Return Format
The query returns PEAK, AVER, or RMS.

Example
The following command sets the time-domain power type to RMS.
:SENSe:TPOWer:SUMMary RMS

The following query returns RMS.
:SENSe:TPOWer:SUMMary?

[:SENSe]:TPOWer:AVERage:COUNt

Syntax
[:SENSe]:TPOWer:AVERage:COUNt <integer>
[:SENSe]:TPOWer:AVERage:COUNt?

Description
Sets the average count of the T-power measurement.
Queries the average count of the T-power measurement.

Parameter

Name	Type	Range	Default
<integer>	Integer	1 to 1000	10

Remarks
This command is only valid when the time-domain power measurement is enabled in GPSA mode.

Return Format
The query returns the average count in integer.

Example
The following command sets the average count to 100.
:SENSe:TPOWer:AVERage:COUNt 100

The following query returns 100.
:SENSe:TPOWer:AVERage:COUNt?

[:SENSe]:TPOWer:AVERage[:STATe]

Syntax
[:SENSe]:TPOWer:AVERage[:STATe] <bool>
[:SENSe]:TPOWer:AVERage[:STATe]?

Description
Enables or disables the average measurement function of the T-power measurement.
Queries the status of the average measurement function of the T-power measurement.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only valid when the time-domain power measurement is enabled in GPSA mode.

Return Format
The query returns 0 or 1.
Example
The following command enables the average measurement.
:SENSe:TPOWer:AVERage:STATe ON or :SENSe:TPOWer:AVERage:STATe 1

The following query returns 1.
:SENSe:TPOWer:AVERage:STATe?

[:SENSe]:TPOWer:AVERage:TCONtrol

Syntax
[:SENSe]:TPOWer:AVERage:TCONtrol <enum>
[:SENSe]:TPOWer:AVERage:TCONtrol?

Description
Sets the average mode of the T-power measurement.
Queries the average mode of the T-power measurement.

Parameter

Name	Type	Range	Default
<enum>	Discrete	EXPonential|REPeat	EXPonential

Remarks
EXPonential: indicates the exponential average.
REPeat: indicates the repeat average.
When "EXPonential" is selected, the result is the exponential average of the measurement results obtained in the past N times (N is specified in *[:SENSe]:TPOWer:AVERage:COUNt*).
When "REPeat" is selected, the result is the arithmetic average of the measurement results obtained in the past N times (N is specified in *[:SENSe]:TPOWer:AVERage:COUNt*).
This command is only valid when the time-domain power measurement is enabled in GPSA mode.

Return Format
The query returns EXP or REP.

Example
The following command sets the average mode to Repeat.
:SENSe:TPOWer:AVERage:TCONtrol REPeat

The following query returns REP.
:SENSe:TPOWer:AVERage:TCONtrol?

[:SENSe]:TPOWer:LLIMit

Syntax
[:SENSe]:TPOWer:LLIMit <time>
[:SENSe]:TPOWer:LLIMit?

Description
Sets the start line for the T-power measurement.
Queries the start line for the T-power measurement.

Parameter

Name	Type	Range	Default
<time>	Real	0 us to (current value of stop line)	0 µs

Remarks
This command is only valid when the time-domain power measurement is enabled in GPSA mode.
Return Format
The query returns the start line in scientific notation. The unit is s.

Example
The following command sets the start line to 5 ms.
:SENSe:TPOWer:LLIMit 0.005

The following query returns 5.000000000e-03.
:SENSe:TPOWer:LLIMit?

[:SENSe]:TPOWer:LLIMit:STATe

Syntax
[:SENSe]:TPOWer:LLIMit:STATe <bool>
[:SENSe]:TPOWer:LLIMit:STATe?

Description
Sets the TP limit state.
Queries the TP limit state.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only valid when the time-domain power measurement is enabled in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the limit line.
:SENSe:TPOWer:LLIMit:STATe 1 or :SENSe:TPOWer:LLIMit:STATe ON

The following query returns 1.
:SENSe:TPOWer:LLIMit:STATe?

[:SENSe]:TPOWer:RLIMit

Syntax
[:SENSe]:TPOWer:RLIMit <time>
[:SENSe]:TPOWer:RLIMit?

Description
Sets the stop line for the T-power measurement.
Queries the stop line for the T-power measurement.

Parameter

Name	Type	Range	Default
<time>	Real	current value of start line to current value of sweep time	1 ms

Remarks

This command is only valid when the time-domain power measurement is enabled in GPSA mode.

Return Format
The query returns the stop line in scientific notation. The unit is s.

Example
The following command sets the stop line to 10 ms.
:SENSe:TPOWer:RLIMit 0.01

The following query returns 1.000000000e-02.
:SENSe:TPOWer:RLIMit?

[:SENSe]:HARMonics:AVERage:COUNt

Syntax
[:SENSe]:HARMonics:AVERage:COUNt <integer>
[:SENSe]:HARMonics:AVERage:COUNt?

Description
Sets the average count of the harmonic distortion measurement.
Queries the average count of the harmonic distortion measurement.

Parameter

Name	Type	Range	Default
<integer>	Integer	1 to 1000	10

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.

Return Format
The query returns the average count in integer.

Example
The following command sets the average count to 100.
:SENSe:HARMonics:AVERage:COUNt 100

The following query returns 100.
:SENSe:HARMonics:AVERage:COUNt?

[:SENSe]:HARMonics:AVERage[:STATe]

Syntax
[:SENSe]:HARMonics:AVERage[:STATe] <bool>
[:SENSe]:HARMonics:AVERage[:STATe]?

Description
Enables or disables the average state of the harmonic distortion measurement.
Queries the average state of the harmonic distortion measurement.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.

Return Format
The query returns 0 or 1.
Example
The following command enables the average measurement function.
:SENSe:HARMonics:AVERage:STATe ON or :SENSe:HARMonics:AVERage:STATe 1

The following query returns 1.
:SENSe:HARMonics:AVERage:STATe?

[:SENSe]:HARMonics:AVERage:TCONtrol

Syntax
[:SENSe]:HARMonics:AVERage:TCONtrol <type>
[:SENSe]:HARMonics:AVERage:TCONtrol?

Description
Sets the average mode of the harmonic distortion measurement.
Queries the average mode of the harmonic distortion measurement.

Parameter

Name	Type	Range	Default
<type>	Discrete	EXPonential|REPeat	EXPonential

Remarks
EXPonential: indicates the exponential average.
REPeat: indicates the repeat average.
When "EXPonential" is selected, the result is the exponential average of the measurement results obtained in the past N times (N is specified in *[:SENSe]:HARMonics:AVERage:COUNt*).
When "REPeat" is selected, the result is the arithmetic average of the measurement results obtained in the past N times (N is specified in [:SENSe]:HARMonics:AVERage:COUNt).
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.

Return Format
The query returns EXP or REP.

Example
The following command sets the average mode to Repeat.
:SENSe:HARMonics:AVERage:TCONtrol REPeat

The following query returns REP.
:SENSe:HARMonics:AVERage:TCONtrol?

[:SENSe]:HARMonics:FREQuency:FUNDamental

Syntax
[:SENSe]:HARMonics:FREQuency:FUNDamental <freq>
[:SENSe]:HARMonics:FREQuency:FUNDamental?

Description
Sets the frequency of the fundamental waveform.
Queries frequency the fundamental waveform.

Parameter

Name	Type	Range	Default
<freq>	Real	0 GHz to 4.25 GHz	1 GHz

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.

Return Format
The query returns the fundamental frequency in scientific notation.

Example
The following command sets the fundamental frequency to 1 GHz.
:SENSe:HARMonics:FREQuency:FUNDamental 1 GHz

The following query returns 1.0E+09.
:SENSe:HARMonics:FREQuency:FUNDamental?

[:SENSe]:HARMonics:FREQuency:FUNDamental:AUTO

Syntax
[:SENSe]:HARMonics:FREQuency:FUNDamental:AUTO <bool>
[:SENSe]:HARMonics:FREQuency:FUNDamental:AUTO?

Description
Sets the auto check of the fundamental waveform.
Queries the auto check of the fundamental waveform.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command sets the auto check of the fundamental waveform.
:SENSe:HARMonics:FREQuency:FUNDamental:AUTO 1
or :SENSe:HARMonics:FREQuency:FUNDamental:AUTO ON

The following query returns 1.
:SENSe:HARMonics:FREQuency:FUNDamental:AUTO?

[:SENSe]:HARMonics:NUMBer

Syntax
[:SENSe]:HARMonics:NUMBer <integer>
[:SENSe]:HARMonics:NUMBer?

Description
Sets No. of Harmo.
Queries No. of Harmo.

Parameter

Name	Type	Range	Default

<integer>	Integer	1 to 10	10

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.

Return Format
The query returns an integer ranging from 1 to 10.

Example
The following command sets the number of harmonics to 5.
:SENSe:HARMonics:NUMBer 5

The following query returns 5.
:SENSe:HARMonics:NUMBer?

[:SENSe]:HARMonics:RTABle:STATe

Syntax
[:SENSe]:HARMonics:RTABle:STATe <bool>
[:SENSe]:HARMonics:RTABle:STATe?

Description
Sets the range table state of the harmonic distortion.
Queries the on/off state of the range table of the harmonic distortion.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the range table of the harmonic distortion.
:SENSe:HARMonics:RTABle:STATe 1 or :SENSe:HARMonics:RTABle:STATe ON

The following query returns 1.
:SENSe:HARMonics:RTABle:STATe?

[:SENSe]:HARMonics:TONE<n>:STATe

Syntax
[:SENSe]:HARMonics:TONE<n>:STATe <bool>
[:SENSe]:HARMonics:TONE<n>:STATe?

Description
Sets the state of measure tone.
Queries the state of measure tone.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8|9|10	——
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.
Return Format
The query returns 0 or 1.

Example
The following command selects the measure tone 1.
:SENSe:HARMonics:TONE1:STATe 1 or :SENSe:HARMonics:TONE1:STATe ON

The following query returns 1.
:SENSe:HARMonics:TONE1:STATe?

[:SENSe]:HARMonics:TONE<n>:FREQuency

Syntax
[:SENSe]:HARMonics:TONE<n>:FREQuency <freq>
[:SENSe]:HARMonics:TONE<n>:FREQuency?

Description
Sets the harmonic frequency.
Queries the harmonic frequency.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8|9|10	——
<freq>	Real	0 GHz to 4.25 GHz	1 GHz

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.

Return Format
The query returns the harmonic frequency in real number.

Example
The following command sets the Harmonic 1 frequency to 2 Hz.
:SENSe:HARMonics:TONE1:FREQuency 2Hz

The following query returns 2.
:SENSe:HARMonics:TONE1:FREQuency?

[:SENSe]:HARMonics:TONE<n>:BANDwidth|BWIDth[:RESolution]

Syntax
[:SENSe]:HARMonics:TONE<n>:BANDwidth|BWIDth[:RESolution] <freq>
[:SENSe]:HARMonics:TONE<n>:BANDwidth|BWIDth[:RESolution]?

Description
Sets the harmonic RBW.
Queries the harmonic RBW.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8|9|10	——
<freq>	Real	0 GHz to 4.25 GHz	1 GHz

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.

Return Format
The query returns the harmonic RBW in real number.

Example
The following command sets the Harmonic 1 RBW value to 2 Hz.
:SENSe:HARMonics:TONE1:BANDwidth:RESolution 2Hz

The following query returns 2.
:SENSe:HARMonics:TONE1:BANDwidth:RESolution?

[:SENSe]:HARMonics:TONE<n>:BANDwidth|BWIDth [:RESolution]:AUTO

Syntax
[:SENSe]:HARMonics:TONE<n>:BANDwidth|BWIDth[:RESolution]:AUTO <bool>
[:SENSe]:HARMonics:TONE<n>:BANDwidth|BWIDth[:RESolution]:AUTO?

Description
Sets the harmonic RBW auto state.
Queries the harmonic RBW auto state.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8|9|10	——
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the Harmonic 1 RBW auto state.
:SENSe:HARMonics:TONE1:BANDwidth:RESolution:AUTO 1
or :SENSe:HARMonics:TONE1:BANDwidth:RESolution:AUTO ON

The following query returns 1.
:SENSe:HARMonics:TONE1:BANDwidth:RESolution:AUTO?

[:SENSe]:HARMonics:TONE<n>:SWEep:TIME

Syntax
[:SENSe]:HARMonics:TONE<n>:SWEep:TIME <freq>
[:SENSe]:HARMonics:TONE<n>:SWEep:TIME?

Description
Sets the dwell time of Harmonic.
Queries the dwell time of Harmonic.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8|9|10	——

<freq>	Real	1 μs to 6,000 s	20 ms

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.

Return Format
The query returns the dwell time of Harmonic in scientific notation.

Example
The following command sets the dwell time of Harmonic 1 to 25 ms.
:SENSe:HARMonics:TONE1:SWEep:TIME 25

The following query returns +2.50000000000000000E+01.
:SENSe:HARMonics:TONE1:SWEep:TIME?

[:SENSe]:HARMonics:TONE<n>:SWEep:TIME:AUTO

Syntax
[:SENSe]:HARMonics:TONE<n>:SWEep:TIME:AUTO <bool>
[:SENSe]:HARMonics:TONE<n>:SWEep:TIME:AUTO?

Description
Sets the dwell time auto state of Harmonic.
Queries the dwell time auto state of Harmonic.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6|7|8|9|10	——
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only valid when the harmonic distortion measurement is enabled in GPSA mode.

Return Format
The query returns 0 or 1.

Example
The following command enables the Harmonic 1 dwell time auto state.
:SENSe:HARMonics:TONE1:SWEep:TIME:AUTO 1 or :SENSe:HARMonics:TONE1:SWEep:TIME:AUTO ON

The following query returns 1.
:SENSe:HARMonics:TONE1:SWEep:TIME:AUTO?

[:SENSe]:MinSigDUR?

Syntax
[:SENSe]:MinSigDUR?

Description
Queries the Min. signal duration for 100% POI.

[:SENSe]:DEMod:STATe

Syntax

[:SENSe]:DEMod:STATe <bool>
[:SENSe]:DEMod:STATe?

Description
Sets or queries the switch status of the audio demodulation function.

Parameter

Name	Type	Range	Default
<bool>	Boolean	OFF|ON|0|1	OFF

Remarks
This command is associated with the mode configuration of the [:SENSe]:DEMod command.。
If the current demodulation mode is OFF and you set the switch state to ON, the demodulation mode will automatically switch to AM mode.

Return Format
The query command returns OFF, ON, 0 or 1.

Example
:SENSe:DEMod:STATe ON /* Enable the audio demodulation function */
:SENSe:DEMod:STATe? /* The query returns 1 */

[:SENSe]:DEMod

Syntax

[:SENSe]:DEMod <mode>
[:SENSe]:DEMod?

Description
Sets or queries the audio demodulation mode.

Parameter

Name	Type	Range	Default
<mode>	Discrete	AM|FM|OFF	OFF

Remarks
When set to OFF, the demodulation status will automatically switch to OFF; when set to AM or FM, the demodulation status will automatically switch to ON.
This command is only available in GPSA mode. When the demodulation mode is AM or FM, demodulation is executed at the absolute frequency of the current marker. If no marker is enabled, the instrument will automatically create one position marker.。

Return Format
The query command returns AM, FM or OFF.

Example
:SENSe:DEMod FM /*Sets the audio demodulation mode to Frequency Modulation (FM) */
:SENSe:DEMod? /*The query returns FM*/

[:SENSe]:DEMod:AUDIo:TIMe

Syntax

[:SENSe]:DEMod:AUDIo:TIMe <value>
[:SENSe]:DEMod:AUDIo:TIMe?

Description
Sets or queries the dwell time of audio demodulation.

Parameter

Name	Type	Range	Default
<value>	Floating-point	5 ms~1000 s	100 ms

Remarks
This command only takes effect when the demodulation function is enabled. The dwell time determines the audio demodulation duration and the trace update rate.

Return Format
The query command returns a floating-point value measured in seconds (s).

Example
:SENSe:DEMod:AUDIo:TIMe 0.1 /* Sets the dwell time to 100 ms */
:SENSe:DEMod:AUDIo:TIMe? /* The query returns 1.0E-01 */

[:SENSe]:DEMod:GAIN:AUTO

Syntax

[:SENSe]:DEMod:GAIN:AUTO <bool>
[:SENSe]:DEMod:GAIN:AUTO?

Description
Sets or queries the automatic gain mode of the demodulated signal.

Parameter

Name	Type	Range	Default
<bool>	Boolean	OFF|ON|0|1	ON

Remarks
ON 为自动增益，OFF 为手动增益。仅当解调功能打开时生效。

Return Format
ON corresponds to automatic gain mode, while OFF corresponds to manual gain mode. This command only takes effect when the demodulation function is enabled.

Example
:SENSe:DEMod:GAIN:AUTO ON /* Enable automatic gain */
:SENSe:DEMod:GAIN:AUTO? /* The query returns ON */

[:SENSe]:DEMod:GAIN:INCRement

Syntax

[:SENSe]:DEMod:GAIN:INCRement <value>
[:SENSe]:DEMod:GAIN:INCRement?

Description
Sets or queries the gain gear of the audio signal.

Parameter

Name	Type	Range	Default
<value>	Integer	1 to 7	7

Remarks
After manually setting this parameter, the gain mode shall be switched to manual mode (OFF).

Return Format
The query command returns an integer gear value in the range of 1 to 7.

Example
:SENSe:DEMod:GAIN:INCRement 5 /* Sets the signal gain gear to 5 */
:SENSe:DEMod:GAIN:INCRement? /* The query returns 5 */

:SOURce Commands

Remarks:
The :SOURce commands are only available in GPSA mode.

:SOURce:CORRection:OFFSet

Syntax
:SOURce:CORRection:OFFSet <rel_ampl>
:SOURce:CORRection:OFFSet?

Description
Sets the offset of the output amplitude of the TG.
Queries the offset of the output amplitude of the tracking generator.

Parameter

Name	Type	Range	Default
<rel_ampl>	Real	-200 dB to 200 dB	0 dB

Return Format
The query returns the offset in scientific notation.
This parameter only changes the readout of the TG output power, but does not affect the actual output power.

Example
The following command sets the offset to 10 dB.
:SOURce:CORRection:OFFSet 10

The following query returns 1.000000000e+01.
:SOURce:CORRection:OFFSet?

:SOURce[:EXTernal]:POWer[:LEVel][:IMMediate][:AMPLitude]

Syntax
:SOURce[:EXTernal]:POWer[:LEVel][:IMMediate][:AMPLitude] <ampl>
:SOURce[:EXTernal]:POWer[:LEVel][:IMMediate][:AMPLitude]?

Description
Sets the output amplitude of the TG.
Queries the output amplitude of the TG.

Parameter

Name	Type	Range	Default
<ampl>	Real	-40 dBm to 0 dBm	-10 dBm

Return Format
The query returns the output amplitude in scientific notation.

Example
The following command sets the output amplitude to -10 dB.
:SOURce:EXTernal:POWer:LEVel:IMMediate:AMPLitude -10

The following query returns -1.000000000e+01.
:SOURce:EXTernal:POWer:LEVel:IMMediate:AMPLitude?

:SOURce:TRACe:REFerence:STATe

Syntax
:SOURce:TRACe:REFerence:STATe <bool>
:SOURce:TRACe:REFerence:STATe?

Description
Sets whether to display the reference trace of normalization.
Queries whether to display the reference trace of normalization.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only valid when the tracking generator function is enabled.

Return Format
The query returns 0 or 1.

Example
The following command enables the reference trace.
:SOURce:TRACe:REF:STATe ON or :SOURce:TRACe:REF:STATe 1

The following query returns 1.
:SOURce:TRACe:REF:STATe?

:SOURce:TRACe:STORref

Syntax
:SOURce:TRACe:STORref

Description
Saves the reference trace of normalization.

:SYSTem Commands

:SYSTem:BEEPer

Syntax
:SYSTem:BEEPer <bool>
:SYSTem:BEEPer?

Description
Enables or disables the beeper; queries the on/off status of the beeper.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 0 or 1.

Example
:SYSTem:BEEPer ON or :SYSTem:BEEPer 1 /*Enables the beeper.*/
:SYSTem:BEEPer? /*The query returns 1.*/

:SYSTem:BRIGhtness

Syntax
:SYSTem:BRIGhtness <brightness>
:SYSTem:BRIGhtness?

Description
Sets or queries the screen brightness.

Parameter

Name	Type	Range	Default
<brightness>	Integer	0% to 100%	80%

Return Format
The query returns the screen brightness in integer.

Example
:SYSTem:BRIGhtness 50 /*Sets the screen brightness to 50%.*/
:SYSTem:BRIGhtness? /*The query returns 50.*/

:SYSTem:DATE

Syntax
:SYSTem:DATE <year>,<month>,<day>
:SYSTem:DATE?

Description
Sets or queries the system date of the instrument.

Parameter

Name	Type	Range	Default
<year>	ASCII String	2000 to 2099	——

<month>	ASCII String	01 to 12	——
<day>	ASCII String	01 to 31	——

Return Format
The query returns the current system date in the format of "YYYY-MM-DD".

Example
:SYSTem:DATE 2017,11,16 /*Sets the system date of the instrument to Nov. 16th, 2017.*/
:SYSTem:DATE? /*The query returns 2017-11-16.*/

:SYSTem:GPIB

Syntax
:SYSTem:GPIB <adr>
:SYSTem:GPIB?

Description
Sets or queries the GPIB address.

Parameter

Name	Type	Range	Default
<adr>	Integer	1 to 30	1

Return Format
The query returns an integer ranging from 1 to 30.

Example
:SYSTem:GPIB 2 /*Sets the GPIB address to 2.*/
:SYSTem:GPIB? /*The query returns 2.*/

:SYSTem:LANGuage

Syntax
:SYSTem:LANGuage <enum>

Description
Sets or queries the system language of the instrument.

Parameter

Name	Type	Range	Default
<enum>	Discrete	SCHinese|TCHinese|KORean|JAPanese|ENGLish|GERMan|PORTuguese|POLish|FRENch|RUSSian|SPAN|THAI|INDonesian	ENGLish

Remarks
SCHinese: Simplified Chinese.
TCHinese: Traditional Chinese.
KORean: Korean.
JAPanese: Japanese.
ENGLish: English.
GERMan: German.
PORTuguese: Portuguese.
POLish: Polish.
FRENch: French.
RUSSian: Russian.

SPAN: Spanish.
THAI: Thai.
INDonesian: Indonesian.

Return Format
The query returns SCH, TCH, KOR, JAP, ENGL, GERM, PORT, POL, FREN, RUSS, SPAN, THAI, or IND.

Example
:SYSTem:LANGuage ENGLish /*Sets the system language to ENGLish.*/
:SYSTem:LANGuage? /*The query returns ENGL.*/

:SYSTem:LOCKed

Syntax
:SYSTem:LOCKed<bool>
:SYSTem:LOCKed?

Description
Locks or unlocks the keypad.
Queries whether the keypad is locked or not.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 0 or 1.

Example
:SYSTem:LOCKed 1 or :SYSTem:LOCKed ON /*Disables the touch screen operation.*/
:SYSTem:LOCKed? /*The query returns 1.*/

:SYSTem:OPTion:INSTall

Syntax
:SYSTem:OPTion:INSTall <option info>@<license info>

Description
Installs and activates the specified option.

Parameter

Name	Type	Range	Default
<option info>	ASCII String	--	--
<license info>	ASCII String	--	--

Remarks
The parameter <option info> indicates the order number of the option. <license info> indicates the serial number of the option.

Example
The following command installs the option RSA800-PNOISE.
:SYSTem:LKEY RSA800-PNOISE @
8AD12B8EBC5DF492D1D4289B7CBA5B6150BF6F5D752D645C36D74530B05F39B49C461B23A50D
6C94A34E06782AC4380070B0D1A86BA84E02768391FFD70C2103

:SYSTem:OPTion:STATus?

Syntax
:SYSTem:OPTion:STATus? <option name>

Description
Queries whether the specified option is activated.

Parameter

Name	Type	Range	Default
<option name>	Discrete	BND|EMI|ADM|VSA|AMK|PNOISE	——

Return Format
The query returns 0 (not activated) or 1 (activated).

Example
The following command queries whether the RSA800- AMK option is activated.
:SYSTem:OPTion:STATus? RSA800-AMK

:SYSTem:OPTion:UNINstall

Syntax
:SYSTem:OPTion:UNINstall

Description
Uninstalls all the official options.

:SYSTem:PON

Syntax
:SYSTem:PON <power_on>
:SYSTem:PON?

Description
Sets or queries the setting type when the instrument is powered on.

Parameter

Name	Type	Range	Default
<power_on>	Discrete	DEFault|LATest	PRESet

Remarks
DEFault: indicates preset settings, including factory mode and 6 user-defined settings.
LATest: last settings.

Return Format
The query returns DEF or LAT.

Example
:SYSTem:PON LAT /*Sets the instrument to recall last settings at next power-on.*/
:SYSTem:PON? /*The query returns LAT.*/

:SYSTem:RESet

Syntax
:SYSTem:RESet

Description
Restarts the instrument.

:SYSTem:PRESet

Syntax
:SYSTem:PRESet

Description
Initializes the instrument.

:SYSTem:PSTatus

Syntax
:SYSTem:PSTatus <status>
:SYSTem:PSTatus?

Description
Sets or queries the front-panel power switch status.

Parameter

Name	Type	Range	Default
<status>	Discrete	DEFault|OPEN	DEFault

Remarks
DEFault: switches off. OPEN: switches on.

Return Format
The query returns DEF or OPEN.

Example
:SYSTem:PSTatus OPEN /*Sets the power switch to be on.*/
:SYSTem:PSTatus? /*The query returns OPEN.*/

:SYSTem:PWRD

Syntax
:SYSTem:PWRD

Description
Powers off the instrument.

:SYSTem:STIME

Syntax
:SYSTem:STIME <bool>
:SYSTem:STIME?

Description
Sets or queries whether to display the system time.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 0 or 1.

Example
:SYSTem:STIME 1 or :SYSTem:STIME ON /*Sets to display the system date.*/
:SYSTem:STIME? /*The query returns 1.*/

:SYSTem:TIME

Syntax
:SYSTem:TIME <hour>,<minute>,<second>
:SYSTem:TIME?

Description
Sets or queries the system time of the instrument.

Parameter

Name	Type	Range	Default
<hour>	ASCII String	00 to 23	——
<minute>	ASCII String	00 to 59	——
<second>	ASCII String	00 to 59	——

Return Format
The query returns the current system time in the format of "HH:MM:SS".

Example
:SYSTem:TIME 15,10,30 /*Sets the system time of the instrument to 15:10:30.*/
:SYSTem:TIME? /*The query returns 15:10:30.*/

:SYSTem:VERSion?

Syntax
:SYSTem:VERSion?

Description
Queries the version number of the SCPI used by the system.

:TRACe Commands

:TRACe:CLEar:ALL

Syntax
:TRACe:CLEar:ALL

Description
Clears all the traces.

:TRACe:SELect

Syntax
:TRACe:SELect <trace>
:TRACe:SELect?

Description
Sets the selected trace.
Queries the selected trace.

Parameter

Name	Type	Range	Default
<trace>	Discrete	TRACe1|TRACe2|TRACe3|TRACe4|TRACe5|TRACe6	TRACe1

Example
The following command clears Trace 2.
:TRACe:SELect TRACe2

The following query returns TRAC2.
:TRACe:SELect?

:TRACe:DISPlay:VIEW:SPECtrogram:TIME?

Syntax
:TRACe:DISPlay:VIEW:SPECtrogram:TIME?

Description
Queries the display trace expressed in time in the Waterfall view when the display mode is Time.

Remarks
This command is only valid in the Spectrogram view in RTSA mode.

:TRACe:PRESet:ALL

Syntax
:TRACe:PRESet:ALL

Description
Resets all the traces.

Remarks
Enables Trace 1, and sets other traces to be in blank state. This does not affect the trace type and the detector type.

:TRACe:MATH:PEAK[:DATA]?

Syntax
:TRACe:MATH:PEAK[:DATA]?

Description
Queries the frequencies (Hz) and amplitudes (the unit is the same as the current Y-axis unit) of the peaks in the peak table.

Remarks
This command only queries Trace 1.

Return Format
The query returns the frequencies and amplitudes of the peaks in the peak table in the following format. At most, 20 pairs of frequencies and amplitudes can be returned.
For example, 1.000000000e+09,-1.463000000e+01,9.999890000e+08,-4.172000000e+01.

:TRACe:MATH:PEAK:POINts?

Syntax
:TRACe:MATH:PEAK:POINts?

Description
Queries the number of peaks in the peak table.

Return Format
The query returns an integer ranging from 0 to 20.

:TRACe[:DATA]

Syntax
:TRACe[:DATA] <enum>,<comma_separated_ASCII_data>
:TRACe[:DATA]? <enum>

Description
Loads the user data to the specified trace.
Queries the log data of the specified trace.

Parameter

Name	Type	Range	Default
<enum>	Discrete	TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6	——
<comma_separated_ASCII_data>	ASCII String	——	——

Remarks
1. <comma_separated_ASCII_data>: indicates the data block[1]. Select this parameter when the data format is ASCii. This parameter uses a comma (1 byte) and a space to separate each data point; each point is expressed in scientific notation. For example, -1.390530e+01 (13 bytes). At most, 801 points can be sent.
2. When VSWR is enabled, the :TRACe? TRACE1 command returns the reference data of VSWR. The :TRACe? TRACE6 command returns the enabled calibrated trace data. The :TRACe? TRACE3 (math operation trace is selected currently) command returns the difference (return loss) between the above two data.

Return Format
The query returns the data of the specified trace.

Example[2]
1. When the data format is **ASCii**,
the following command sends the user data to Trace1.
:TRACe:DATA TRACE1, -1.390530e+01, -7.108871e+01, -7.089631e+01, -6.992984e+01, -7.010770e+01, ……

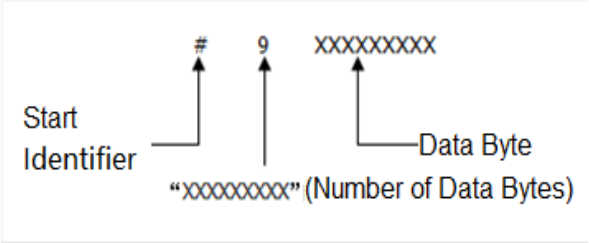
The following query returns -1.390530e+01, -7.108871e+01, -7.089631e+01, -6.992984e+01, -7.010770e+01, ……
:TRACe:DATA? TRACE1

2. When the data format is REAL[,32],
the following command sends the user data to Trace2.
:TRACe:DATA TRACE2,
#900000320411111111111111111111111111100111111111111111111111111111111011101111111111111111111111010011……

The following query returns
#900000320411111111111111111111111111100111111111111111111111111111111011101111111111111111111111010011……
:TRACe:DATA? TRACE2

Note:

1. The PC software converts the returned binary data to ASCII format. Therefore, if you use the PC software to perform the query command, the returned value is displayed as a messy code.
2. The format of data block is "data block header+data block". Wherein, the format of the data block header is as follows:



The first figure (9) following # indicates the remaining number of figures in the data block header. The rest figures in the data block header indicate the number of data bytes for this transmission (if the number of data bytes is less than 9, 0 is added ahead of the number). For example, if you send 100-byte data, its data block header should be #9000000100.

3. The two examples in this page take sending 801 points as an example, and the query returns 801 points. Due to the limited space, the subsequent data are omitted. In addition, the returned values of the above examples are obtained when the system stops sweeping the trace.

:TRACe<n>:TYPE

Syntax
:TRACe<n>:TYPE <enum>
:TRACe<n>:TYPE?

Description
Sets the type of the specified trace.
Queries the type of the specified trace.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	——
<enum>	Discrete	WRITe|AVERage|MAXHold|MINHold	WRITe

Remarks
WRITe: indicates clear/write.
AVERage: indicates average.
MAXHold: indicates the maximum hold.
MINHold: indicates the minimum hold.
Only Trace 1 is available in the view that includes Density.
In the view that includes PvT, only Trace 1 is available and the trace type can only be set to "Clear Write".

Return Format
The query returns WRIT, AVER, MAXH, or MINH.

Example
The following command sets the type of Trace 1 to Max Hold.
:TRACe1:TYPE MAXHold

The following query returns MAXH.
:TRACe1:TYPE?

:TRACe<n>:DISPlay:STATe

Syntax
:TRACe<n>:DISPlay[:STATe] <bool>
:TRACe<n>:DISPlay[:STATe]?

Description
Enables or disables the display of the specified trace.
Queries the setting state for the specified trace.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	——
<bool>	Bool	OFF|ON|0|1	Refer to "Remarks"

Remarks
By default, Trace 1 is enabled (ON), and Trace 2 through Trace 6 are disabled (OFF).

Return Format
The query returns 1 or 0.

Example
The following command enables the display of Trace 1.
:TRACe1:DISPlay:STATe ON or :TRACe1:DISPlay:STATe 1

The following query returns 1.
:TRACe1:DISPlay:STATe?

:TRACe<n>:UPDate:STATe

Syntax
:TRACe<n>:UPDate:STATe <bool>
:TRACe<n>:UPDate:STATe?

Description
Enables or disables the update of the specified trace.
Queries the setting state for the update of the specified trace.
Parameter

Name	Type	Range	Default
<n>	Discrete	1|2|3|4|5|6	——
<bool>	Bool	OFF|ON|0|1	Refer to "Remarks"

Remarks
By default, Trace 1 is enabled (ON), and Trace 2 through Trace 6 are disabled (OFF).

Return Format

The query returns 1 or 0.

Example
The following command enables the update of Trace 1.
:TRACe1:UPDate:STATe ON or :TRACe1:UPDate:STATe 1

The following query returns 1.
:TRACe1:UPDate:STATe?

:TRIGger Commands

:TRIGger[:SEQuence]:SOURce

Syntax
:TRIGger[:SEQuence]:SOURce <type>
:TRIGger[:SEQuence]:SOURce?

Description
Sets the trigger source.
Queries the trigger source.

Parameter

Name	Type	Range	Default
<type>	Discrete	IMMediate|EXTernal|VIDeo|POWer|FMT	IMMediate

Remarks
- The valid parameters in GPSA mode include IMMediate, EXTernal, and VIDeo.
  IMMediate: indicates the free-run trigger.
  EXTernal: indicates the external trigger.
  VIDeo: Video trigger.
- The valid parameters in RTSA mode include IMMediate, EXTernal, POWer, and FMT.
  IMMediate: indicates the free-run trigger.
  EXTernal: indicates the external trigger.
  POWer: indicates the IF power trigger.
  FMT: FMT trigger

Return Format
The query returns IMM, EXT1, VID, POW, or FMT.

Example
The following command sets the trigger source to external trigger.
:TRIGger:SEQuence:SOURce EXTernal

The following query returns EXT1.
:TRIGger:SEQuence:SOURce?

:TRIGger[:SEQuence]:ATRigger

Syntax
:TRIGger[:SEQuence]:ATRigger <time>
:TRIGger[:SEQuence]:ATRigger?

Description
Sets the time that the analyzer will wait for the trigger to be initiated automatically.
Queries the time that the analyzer will wait for the trigger to be initiated automatically.

Parameter

Name	Type	Range	Default
<time>	Real	1 ms to 100 s	100 ms

Remarks
This command is only valid when the auto triggering function is enabled.

Return Format
The query returns the time value in scientific notation. The unit is s.

Example
The following command sets the time to 10 ms.
:TRIGger:SEQuence:ATRigger 0.01

The following query returns 1.000000000e-02.
:TRIGger:SEQuence:ATRigger?

:TRIGger[:SEQuence]:ATRigger:STATe

Syntax
:TRIGger[:SEQuence]:ATRigger:STATe <bool>
:TRIGger[:SEQuence]:ATRigger:STATe?

Description
Enables or disables the auto trigger function.
Queries the setting status of auto trigger function.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 1 or 0.

Example
The following command enables the auto trigger function.
:TRIGger:SEQuence:ATRigger:STATe ON or :TRIGger:SEQuence:ATRigger:STATe 1

The following query returns 1.
:TRIGger:SEQuence:ATRigger:STATe?

:TRIGger<n>[:SEQuence]:EXTernal:DELay

Syntax
:TRIGger<n>[:SEQuence]:EXTernal:DELay <time>
:TRIGger<n>[:SEQuence]:EXTernal:DELay?

Description
Sets the delay time for the external trigger.
Queries the delay time for the external trigger.

Parameter

Name	Type	Range	Default
<n>	Discrete	1	1
<time>	Real	0 us to 500 ms	1 us

Remarks
This command is only valid when the external trigger delay function is enabled. The parameter <n> can be omitted.

Return Format
The query returns the delay time for the external trigger in scientific notation. The unit is s.

Example
The following command sets the delay time for External Trigger to 100 ms.
:TRIGger:SEQuence:EXTernal:DELay 0.1

The following query returns 1.000000000e-01.
:TRIGger:SEQuence:EXTernal:DELay?

:TRIGger<n>[:SEQuence]:EXTernal:DELay:STATe

Syntax
:TRIGger<n>[:SEQuence]:EXTernal:DELay:STATe <bool>
:TRIGger<n>[:SEQuence]:EXTernal:DELay:STATe?

Description
Enables or disables the external trigger delay function.
Queries the setting state of the external trigger delay function.

Parameter

Name	Type	Range	Default
<n>	Discrete	1	1
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
The parameter <n> can be omitted.

Return Format
The query returns 1 or 0.

Example
The following command enables the delay function of External Trigger.
:TRIGger:SEQuence:EXTernal:DELay:STATe ON or :TRIGger:SEQuence:EXTernal:DELay:STATe 1

The following query returns 1.
:TRIGger:SEQuence:EXTernal:DELay:STATe?

:TRIGger<n>[:SEQuence]:EXTernal:SLOPe

Syntax
:TRIGger<n>:SEQuence:EXTernal:SLOPe <type>
:TRIGger<n>:SEQuence:EXTernal:SLOPe?

Description
Sets the trigger edge for the external trigger.
Queries the trigger edge for the external trigger.

Parameter

Name	Type	Range	Default
<n>	Discrete	1	1
<type>	Discrete	POSitive|NEGative	POSitive

Remarks
POSitive: indicates the rising edge.
NEGative: indicates the falling edge.
The parameter <n> can be omitted.

Return Format
The query returns POS or NEG.

Example
The following command sets the trigger edge of External Trigger to POSitive.

:TRIGger:SEQuence:EXTernal:SLOPe POSitive

The following query returns POS.
:TRIGger:SEQuence:EXTernal:SLOPe?

:TRIGger<n>[:SEQuence]:OUTPut

Syntax
:TRIGger<n>:SEQuence:OUTPut <bool>
:TRIGger<n>:SEQuence:OUTPut?

Description
Enables or disables external trigger output。
Queries whether the external trigger output is enabled or not.

Parameter

Name	Type	Range	Default
<n>	Discrete	1	1
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
The parameter <n> can be omitted.

Return Format
The query returns 0 or 1.

Example
The following command enables the external trigger output.
:TRIGger:SEQuence:OUTPut 1 or :TRIGger:SEQuence:OUTPut ON

The following query returns 1.
:TRIGger:SEQuence:OUTPut?

:TRIGger<n>[:SEQuence]:OUTPut:POLarity

Syntax
:TRIGger<n>:SEQuence:OUTPut:POLarity <type>
:TRIGger<n>:SEQuence:OUTPut:POLarity?

Description
Sets the polarity of the external trigger output.
Queries the polarity of the external trigger output.

Parameter

Name	Type	Range	Default
<n>	Discrete	1	1
<type>	Discrete	POSitive|NEGative	POSitive

Remarks
POSitive: indicates positive polarity.
NEGative: indicates negative polarity.
The parameter <n> can be omitted.

Return Format
The query returns POS or NEG.

Example
The following command sets the output polarity of the external trigger output to POSitive.
:TRIGger:SEQuence:OUTPut:POLarity POSitive

The following query returns POS.
:TRIGger:SEQuence:OUTPut:POLarity?

:TRIGger[:SEQuence]:IF:DELay

Syntax
:TRIGger[:SEQuence]:IF:DELay <time>
:TRIGger[:SEQuence]:IF:DELay?

Description
Sets the delay time for the IF power trigger.
Queries the delay time for the IF power trigger.

Parameter

Name	Type	Range	Default
<time>	Real	1 ms to 100 s	100 ms

Remarks
This command is only valid when the IF power trigger function is enabled.

Return Format
The query returns the time value in scientific notation. The unit is s.

Example
The following command sets the delay time for the IF power trigger to 10 ms.
:TRIGger:SEQuence:IF:DELay 0.01

The following query returns 1.000000000e-02.
:TRIGger:SEQuence:IF:DELay?

:TRIGger[:SEQuence]:IF:DELay:STATe

Syntax
:TRIGger[:SEQuence]:IF:DELay:STATe <bool>
:TRIGger[:SEQuence]:IF:DELay:STATe?

Description
Sets the delay state for the IF power trigger.
Queries the delay state for the IF power trigger.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 1 or 0.

Example
The following command enables the delay function for the IF power trigger.
:TRIGger:SEQuence:IF:DELay:STATe ON or :TRIGger:SEQuence:IF:DELay:STATe 1

The following query returns 1.

:TRIGger:SEQuence:IF:DELay:STATe?

:TRIGger[:SEQuence]:IF:LEVel

Syntax
:TRIGger[:SEQuence]:IF:LEVel <time>
:TRIGger[:SEQuence]:IF:LEVel?

Description
Sets the trigger level of the IF power trigger.
Queries the trigger level of the IF power trigger.

Parameter

Name	Type	Range	Default
<time>	Real	(-140+Level Offset) to (30+Level Offset)	-25 dBm

Remarks
This command is only valid when the IF power trigger function is enabled.

Return Format
The query returns the trigger level of the IF power trigger in scientific notation.

Example
The following command sets the trigger level of IF power trigger to 10 dBm.
:TRIGger:SEQuence:IF:LEVel 10

The following query returns 1.000000000e+01.
:TRIGger:SEQuence:IF:LEVel?

:TRIGger[:SEQuence]:FMT:APTRigger

Syntax
:TRIGger[:SEQuence]:FMT:APTRigger <number>
:TRIGger[:SEQuence]:FMT:APTRigger?

Description
Sets the number of times for acquisition after each effective trigger is completed.
Queries the number of times for acquisition after each effective trigger is completed.

Parameter

Name	Type	Range	Default
<number>	Integer	Refer to "Remarks"	1

Remarks
The minimum value of the parameter <number> is 1. When the trigger criteria of FMT is "Inside" or "Outside", the maximum value of the parameter <number> is 1. In the Spectrogram measurement mode, the maximum value of the parameter <number> is 10,000. In the PvT measurement mode, the maximum value of the parameter <number> is 5,000.
This command is only valid when you select FMT to be the trigger source in RTSA working mode.

Return Format
The query returns the number of acquisition times in integer.

Example
The following command sets the number of acquisition times to 5.
:TRIGger:SEQuence:FMT:APTRigger 5
The following query returns 5.

:TRIGger:SEQuence:FMT:APTRigger?

:TRIGger[:SEQuence]:FMT:CRITeria

Syntax
:TRIGger[:SEQuence]:FMT:CRITeria <type>
:TRIGger[:SEQuence]:FMT:CRITeria?

Description
Sets the trigger criteria for FMT.
Queries the trigger criteria for FMT.

Parameter

Name	Type	Range	Default
<type>	Discrete	ENTer|LEAVe|INSide|OUTSide|ELEave|LENTer	ENTer

Remarks
This command is only valid when you select FMT to be the trigger source in RTSA working mode.

Return Format
The query returns ENT, LEAV, INS, OUTS, ELE, or LENT.

Example
The following command sets the trigger criteria for FMT to Enter.
:TRIGger:SEQuence:FMT:CRITeria ENTer

The following query returns ENT.
:TRIGger:SEQuence:FMT:CRITeriar?

:TRIGger[:SEQuence]:FMT:DELay

Syntax
:TRIGger[:SEQuence]:FMT:DELay <time>
:TRIGger[:SEQuence]:FMT:DELay?

Description
Sets the delay time for FMT.
Queries the delay time for FMT.

Parameter

Name	Type	Range	Default
<time>	Real	0 us to 500 ms	1 us

Remarks

When "Inside" or "Outside" is selected to be the trigger criteria, the FMT trigger delay command is invalid.
This command is only valid when the following conditions are met: FMT is selected to be the trigger source in RTSA working mode; FMT trigger delay function is enabled.

Return Format
The query returns the delay time for FMT in scientific notation. The unit is s.

Example
The following command sets the delay time for FMT to 100 ms.
:TRIGger:SEQuence:FMT:DELay 0.1

The following query returns 1.000000000e-01.
:TRIGger:SEQuence:FMT:DELay?

:TRIGger[:SEQuence]:FMT:DELay:STATe

Syntax
:TRIGger[:SEQuence]:FMT:DELay:STATe <bool>
:TRIGger[:SEQuence]:FMT:DELay:STATe?

Description
Enables or disables the FMT trigger delay function.
Queries the status of the FMT trigger delay function.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
This command is only valid when you select FMT to be the trigger source in RTSA working mode.

Return Format
The query returns 1 or 0.

Example
The following command enables the FMT trigger delay function.
:TRIGger:SEQuence:FMT:DELay:STATe ON or :TRIGger:SEQuence:FMT:DELay:STATe 1

The following query returns 1.
:TRIGger:SEQuence:FMT:DELay:STATe?

:TRIGger[:SEQuence]:FMT:MASK

Syntax
:TRIGger[:SEQuence]:FMT:MASK <type>
:TRIGger[:SEQuence]:FMT:MASK?

Description
Sets the mask to be used for the current trigger.
Queries the mask to be used for the current trigger.

Parameter

Name	Type	Range	Default
<type>	Discrete	UPPer|LOWer|BOTH	UPPer

Remarks
This command is only valid when you select FMT to be the trigger source in RTSA working mode.

Return Format
The query returns UPP, LOW, or BOTH.

Example
The following command sets the mask to be used for the current trigger to "Upper".
:TRIGger:SEQuence:FMT:MASK UPPer

The following query returns UPP.
:TRIGger:SEQuence:FMT:MASK?

:TRIGger[:SEQuence]:FMT:MASK:EDIT

Syntax
:TRIGger[:SEQuence]:FMT:MASK:EDIT <type>
:TRIGger[:SEQuence]:FMT:MASK:EDIT?

Description
Sets the mask type that is currently viewed/edited.
Queries the mask type that is currently viewed/edited.

Parameter

Name	Type	Range	Default
<type>	Discrete	UPPer|LOWer	UPPer

Remarks
This command is only valid when you select FMT to be the trigger source in RTSA working mode.

Return Format
The query returns UPP or LOW.

Example
The following command sets the mask that is currently viewed/edited to "Upper".
:TRIGger:SEQuence:FMT:MASK:EDIT UPPer

The following query returns UPP.
:TRIGger:SEQuence:FMT:MASK:EDIT?

:TRIGger[:SEQuence]:FMT:MASK:DELete:ALL

Syntax
:TRIGger[:SEQuence]:FMT:MASK:DELete:ALL

Description
Deletes all the masks.

:TRIGger[:SEQuence]:FMT:MASK:RELative:AMPLitude

Syntax
:TRIGger[:SEQuence]:FMT:MASK:RELative:AMPLitude <bool>
:TRIGger[:SEQuence]:FMT:MASK:RELative:AMPLitude?

Description
Sets the Y-axis type of FMT.
Queries the Y-axis type of FMT.

Parameter

Name	Type	Range	Default
<bool>	Bool	ON|OFF|1|0	OFF|0

Remarks
OFF|0: disables the coupling function. That is, "Fixed" is selected under **Y Axis Type**, and the amplitude of the current mask point is not affected by the reference level.
ON|1: enables the coupling function. When "Relative" is selected under **Y Axis Type**, the amplitude of the current mask point is the difference between the amplitude of the current mask point and the current reference level. If you change the parameter state after completing the mask setup, the amplitude of the mask should change with the state to keep the mask points to be in the same position relative to the current reference level

of the instrument.
When enabled, the amplitudes of the mask points are expressed as an offset from the reference level.
This command is only valid when you select FMT to be the trigger source in RTSA working mode.

Return Format
The query returns 1 or 0.

Example
The following command enables the coupling relationship between the amplitudes of the mask points and the instrument's reference level.
:TRIGger:SEQuence:FMT:MASK:RELative:AMPLitude ON
or :TRIGger:SEQuence:FMT:MASK:RELative:AMPLitude 1

The following query returns 1.
:TRIGger:SEQuence:FMT:MASK:RELative:AMPLitude?

:TRIGger[:SEQuence]:FMT:MASK:RELative:FREQuency

Syntax
:TRIGger[:SEQuence]:FMT:MASK:RELative:FREQuency <bool>
:TRIGger[:SEQuence]:FMT:MASK:RELative:FREQuency?

Description
Sets the X-axis type of FMT.
Queries the X-axis type of FMT.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
OFF|0: disables the coupling function. That is, "Fixed" is selected under **X Axis Type**, and the frequency of the current mask point is not affected by the center frequency.
ON|1: enables the coupling function. When "Relative" is selected under **X Axis Type**, the amplitude of the current mask point is the difference between the amplitude of the current mask point and the current Center Freq. If you change the parameter state after completing the mask setup, the frequency of the mask should change with the state to keep the mask points to be in the same position relative to the current center frequency of the instrument.
When enabled, the frequencies of the mask points are expressed as an offset from the center frequency. At this time, the frequencies of the mask points can be negative values.
This command is only valid when you select FMT to be the trigger source in RTSA working mode.

Return Format
The query returns 1 or 0.

Example
The following command enables the coupling relationship between the frequencies of the mask points and the instrument's center frequency.
:TRIGger:SEQuence:FMT:MASK:RELative:FREQuency ON
or :TRIGger:SEQuence:FMT:MASK:RELative:FREQuency 1

The following query returns 1.
:TRIGger:SEQuence:FMT:MASK:RELative:FREQuency?

:TRIGger[:SEQuence]:FMT:MASK<n>:NAME

Syntax
:TRIGger[:SEQuence]:FMT:MASK<n>:NAME <string>
:TRIGger[:SEQuence]:FMT:MASK<n>:NAME?

Description
Sets the name of the mask.
Queries the name of the mask.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2	——
<string>	ASCII String	The label can contain English letters and numbers, as well as some symbols.	——

Remarks
When <n> is set to 1, it indicates that the upper mask is being edited; when it is set to 2, it indicates that the lower mask is being edited.
This command is only valid when you select FMT to be the trigger source in RTSA working mode.

Example
The following command sets the mask name on Trace 1 to Rigol.
:TRIGger:SEQuence:FMT:MASK1:NAME Rigol

The following query returns Rigol.
:TRIGger:SEQuence:FMT:MASK1:NAME?

:TRIGger[:SEQuence]:FMT:MASK<n>:BUILd

Syntax
:TRIGger[:SEQuence]:FMT:MASK<n>:BUILd <id>

Description
Creates a mask from a trace.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2	——
<id>	Discrete	TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6	——

Remarks
When <n> is set to 1, it indicates that the upper mask is being edited; when it is set to 2, it indicates that the lower mask is being edited.
This command is only valid when you select FMT to be the trigger source in RTSA working mode.

Example
The following command builds an upper mask from Trace1.
:TRIGger:SEQuence:FMT:MASK1:BUILd TRACE1

:TRIGger[:SEQuence]:FMT:MASK<n>:DATA

Syntax
:TRIGger[:SEQuence]:FMT:MASK<n>:DATA <freq>,<ampl>{,<freq>, <ampl>}
:TRIGger[:SEQuence]:FMT:MASK<n>:DATA?

Description
Edits the mask parameters.
Queries the mask parameters.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2	——
<freq>	Real	0 Hz to 6.5 GHz	——
<ampl>	Real	-1000 dBm to 1000 dBm	——

Remarks
When <n> is set to 1, it indicates that the upper mask is being edited; when it is set to 2, it indicates that the lower mask is being edited.
This command is only valid when you select FMT to be the trigger source in RTSA working mode.

Return Format
The query returns the mask parameters that are being edited currently in scientific notation.

Example
The following command edits the two points of the upper mask.
:TRIGger:SEQuence:FMT:MASK1:DATA –80000000, –50, 80000000, –50

The following query returns –8.000000000e+07, –9.988000000e+01, 8.000000e+07, –9.988000000e+01
:TRIGger:SEQuence:FMT:MASK1:DATA?

:TRIGger[:SEQuence]:FMT:MASK<n>:DELete

Syntax
:TRIGger[:SEQuence]:FMT:MASK<n>:DELete

Description
Deletes the specified mask.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2	——

Remarks
When <n> is set to 1, it indicates that the upper mask is being edited; when it is set to 2, it indicates that the lower mask is being edited.
This command is only valid when you select FMT to be the trigger source in RTSA working mode.

:TRIGger[:SEQuence]:FMT:MASK<n>:NEW

Syntax
:TRIGger[:SEQuence]:FMT:MASK<n>:NEW

Description
Clears the currently activated mask and creates a default mask.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2	——

Remarks
When <n> is set to 1, it indicates that the upper mask is being edited; when it is set to 2, it indicates that the

lower mask is being edited.
This command is only valid when you select FMT to be the trigger source in RTSA working mode.

:TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:X

Syntax
:TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:X <offset>
:TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:X?

Description
Sets the mask frequency offset.
Queries the mask frequency offset.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2	——
<offset>	Real	-1 PHz to 1 PHz	0 Hz

Remarks
When <n> is set to 1, it indicates that the upper mask is being edited; when it is set to 2, it indicates that the lower mask is being edited.
This command is only valid when you select FMT to be the trigger source in RTSA working mode.

Example
The following command sets the the upper mask frequency offset to 1 Hz.
:TRIGger:SEQuence:FMT:MASK1:OFFSet:X 1 Hz

The following query returns 1.
:TRIGger:SEQuence:FMT:MASK1:OFFSet:X?

:TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:Y

Syntax
:TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:Y <offset>
:TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:Y?

Description
Sets the mask amplitude offset.
Queries the mask amplitude offset.

Parameter

Name	Type	Range	Default
<n>	Discrete	1|2	——
<offset>	Real	-	0 dBm

Remarks
When <n> is set to 1, it indicates that the upper mask is being edited; when it is set to 2, it indicates that the lower mask is being edited.
This command is only valid when you select FMT to be the trigger source in RTSA working mode.

Example
The following command sets the upper mask amplitude offset to 1 dBm.
:TRIGger:SEQuence:FMT:MASK1:OFFSet:Y 1 dBm

The following query returns 1.
:TRIGger:SEQuence:FMT:MASK1:OFFSet:Y?

:TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:UPDate

Syntax
:TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:UPDate

Description
Update the offset.

:TRIGger[:SEQuence]:HOLDoff

Syntax
:TRIGger[:SEQuence]:HOLDoff <time>
:TRIGger[:SEQuence]:HOLDoff?

Description
Sets the trigger holdoff time.
Queries the trigger holdoff time.

Parameter

Name	Type	Range	Default
<time>	Real	100 us to 500 ms (GPSA mode) 0 us to 10 s (RTSA mode)	100 ms

Remarks
This command is only valid when the trigger holdoff function is enabled.
Return Format
The query returns the trigger holdoff time in scientific notation. The unit is s.

Example
The following command sets the sync holdoff time to 100 ms.
:TRIGger:SEQuence:HOLDoff 0.1

The following query returns 1.000000000e-01.
:TRIGger:SEQuence:HOLDoff?

:TRIGger[:SEQuence]:HOLDoff:STATe

Syntax
:TRIGger[:SEQuence]:HOLDoff:STATe <bool>
:TRIGger[:SEQuence]:HOLDoff:STATe?

Description
Turns on or off the trigger holdoff function.
Queries the status of the trigger holdoff function.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Return Format
The query returns 1 or 0.

Example
The following command enables the trigger holdoff function.
:TRIGger:SEQuence:HOLDoff:STATe ON or :TRIGger:SEQuence:HOLDoff:STATe 1

The following query returns 1.
:TRIGger:SEQuence:HOLDoff:STATe?

:TRIGger[:SEQuence]:VIDeo:SLOPe

Syntax
:TRIGger[:SEQuence]:VIDeo:SLOPe <type>
:TRIGger[:SEQuence]:VIDeo:SLOPe?

Description
Sets the polarity of Video trigger.
Queries the polarity of Video trigger.

Parameter

Name	Type	Range	Default
<type>	Discrete	POSitive|NEGative	POSitive

Remarks
POSitive: indicates the positive slew rate.
NEGative: indicates the negative slew rate.
The commands are only valid in GPSA mode.

Return Format
The query returns POS or NEG.

Example
The following command sets the polarity of the video trigger to POSitive.
:TRIGger:SEQuence:VIDeo:SLOPe POSitive

The following query returns POS.
:TRIGger:SEQuence:VIDeo:SLOPe?

:TRIGger[:SEQuence]:VIDeo:DELay

Syntax
:TRIGger[:SEQuence]:VIDeo:DELay <time>
:TRIGger[:SEQuence]:VIDeo:DELay?

Description
Sets the delay time for the video trigger.
Queries the delay time for the video trigger.

Parameter

Name	Type	Range	Default
<time>	Real	0 us to 500 ms	1 us

Remarks
This command is only valid when the video trigger delay function is enabled in GPSA working mode.

Return Format
The query returns the delay time for the video trigger in scientific notation. The unit is s.

Example
The following command sets the delay time for the video trigger to 100 ms.
:TRIGger:SEQuence:VIDeo:DELay 0.1

The following query returns 1.000000e-01.
:TRIGger:SEQuence:VIDeo:DELay?

:TRIGger[:SEQuence]:VIDeo:DELay:STATe

Syntax
:TRIGger[:SEQuence]:VIDeo:DELay:STATe <bool>
:TRIGger[:SEQuence]:VIDeo:DELay:STATe?

Description
Enables or disables the video trigger delay function.
Queries the setting state of the video trigger delay function.

Parameter

Name	Type	Range	Default
<bool>	Bool	OFF|ON|0|1	OFF|0

Remarks
The commands are only valid in GPSA mode.

Return Format
The query returns 1 or 0.

Example
The following command enables the delay function for the video trigger.
:TRIGger:SEQuence:VIDeo:DELay:STATe ON or :TRIGger:SEQuence:VIDeo:DELay:STATe 1

The following query returns 1.
:TRIGger:SEQuence:VIDeo:DELay:STATe?

:TRIGger[:SEQuence]:VIDeo:LEVel

Syntax
:TRIGger[:SEQuence]:VIDeo:LEVel <ampl>
:TRIGger[:SEQuence]:VIDeo:LEVel?

Description
Sets the trigger level of Video trigger.
Queries the trigger level of Video trigger.

Parameter

Name	Type	Range	Default
<ampl>	Real	-140 dBm to 30 dBm	-25 dBm

Remarks
This command is only valid when the following conditions are met: in GPSA working mode; selecting the video trigger.

Return Format
The query returns the trigger level in scientific notation.

Example
The following command sets the trigger level of Video trigger to 10 dBm.
:TRIGger:SEQuence:VIDeo:LEVel 10

The following query returns 1.000000e+01.
:TRIGger:SEQuence:VIDeo:LEVel?

:UNIT Commands

:UNIT:POWer

Syntax
:UNIT:POWer <unit>
:UNIT:POWer?

Description
Sets the unit of the Y axis.
Queries the unit of the Y axis.

Parameter

Name	Type	Range	Default
<unit>	Discrete	DBM|DBMV|DBUV|V|W	Refer to "**Remarks**"

Remarks
The default log unit is dBm.
The default linear unit is V.

Return Format
The query returns DBM, DBMV, DBUV, V, or W.

Example
The following command sets the amplitude unit to Watts.
:UNIT:POWer W

The following query returns W.
:UNIT:POWer?

3. Programming Examples

This chapter lists some programming examples to illustrate how to use commands to realize the common functions of the spectrum analyzer in the development environments such as Visual C++ 6.0, Visual Basic 6.0, and LabVIEW 2010. Also, the chapter lists some examples to illustrate how to control the spectrum analyzer to realize the common functions in Linux operating system. These examples are programmed based on NI-VISA library.

NI-VISA (National Instrument-Virtual Instrument Software Architecture), developed by NI (National Instrument), provides an advanced programming interface to communicate with various instruments through their bus lines. NI-VISA enables you to realize the communication between the analyzer and PC through instrument buses (e.g. USB). VISA defines a set of software commands with which users can control the instrument without the need to understand how the interface bus works. For details, please refer to the NI-VISA Help.

Programming Instructions

This section introduces the problems that might occur during the programming process as well as their solutions. If these problems occur, please resolve them according to the corresponding instructions.

1. When you build a working environment via the network, it is recommended that you build a pure local area network.

2. If the local area network environment is complicated (e.g. many devices and broadcast messages exist), it is recommended that you add some fault tolerance during the programming process. For the details, refer to the instrument write/read operations with exception handling functions "InstrWriteEx()" and "InstrReadEx()" in "*Visual C++ 6.0 Programming Example*".

3. The Socket programming port No. of this device is 5025.

Programming Preparations

The programming preparations introduced here are only applicable to programming by using Visual C++ 6.0, Visual Basic 6.0, and LabVIEW 2010 development tools in Windows operating system. For the preparations of programming in Linux operating system, refer to "*Linux* Programming Example" in "*Programming Preparations*".

First, check whether your PC has installed NI's VISA library. If not, download it from *http://www.ni.com/visa/*. In this manual, the default installation path is C:\Program Files\IVI Foundation\VISA.
Connect spectrum analyzer to the PC via the USB interface of the analyzer. Use the USB cable to connect the USB DEVICE interface on the rear panel of the analyzer to the USB interface of the PC.

After the analyzer is connected to the PC properly, start the analyzer. In this case, "Found New Hardware Wizard" dialog box appears on the PC. Please install "USB Test and Measurement Device (IVI)" according to the wizard.

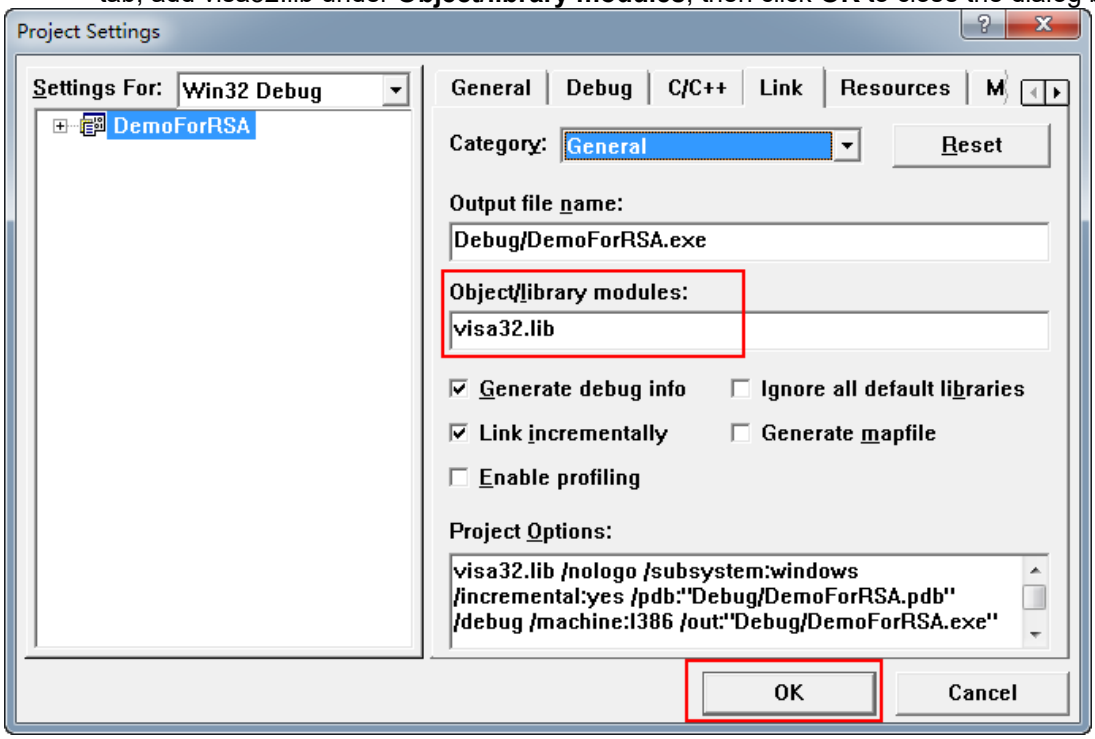
By now, the programming preparations are complete. The following parts will make a detailed introduction about the programming instances in the Visual C++ 6.0, Visual Basic 6.0, and LabVIEW 2010 development environment.

Visual C++ 6.0 Programming Example

Enter the Visual C++6.0 programming environment, and perform the following procedures.

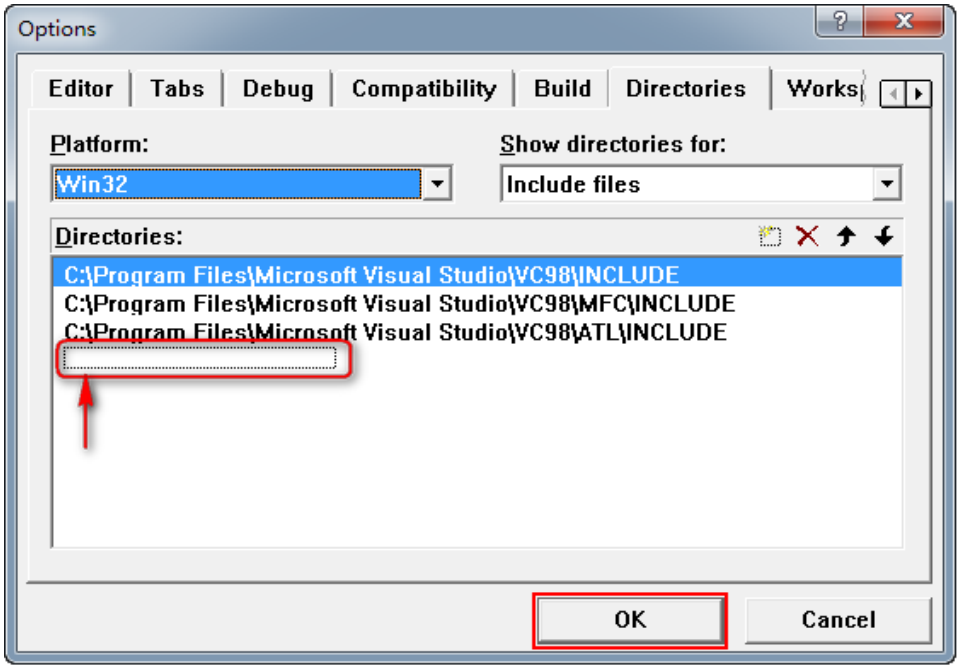
1. Create a MFC project based on a dialog box and name it "DemoForDSA" in this example.

2. Click **Project** → **Settings** to open the Project Setting dialog box. In the dialog box, click the **Link** tab, add visa32.lib under **Object/library modules**, then click **OK** to close the dialog box.



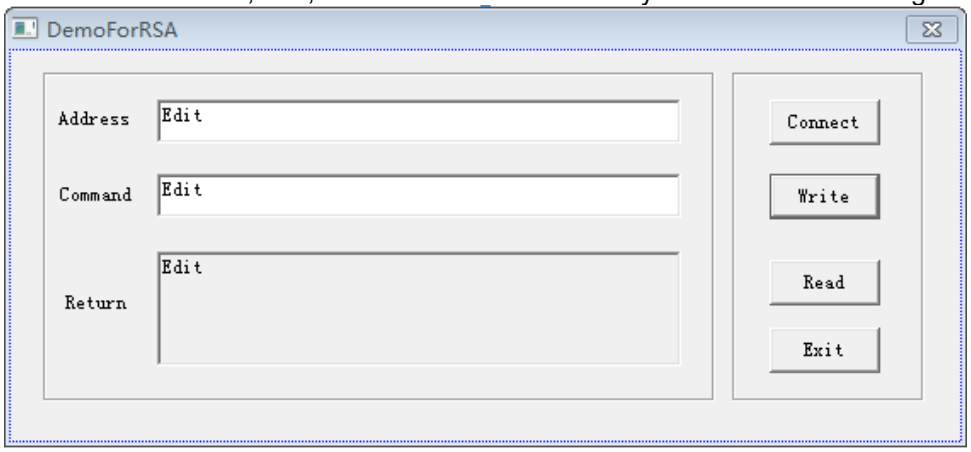
3. Click **Tools** → **Options** to open the **Options** dialog box. Then, click the **Directories** tab.
Select **Include files** from the drop-down list under **Show directories for**. Double click the empty space under **Directories** to enter the specified path of **Include files**: C:\Program Files\IVI Foundation\VISA\WinNT\include. Click **OK** to close the dialog box.
Select Library files from the drop-down list under Show directories for. Double click the empty space under Directories to enter the specified path of Library files：C:\Program Files\IVI Foundation\VISA\WinNT\lib\msc. Click OK to close the dialog box.

Note: The two paths added here are related to the installation path of NI-VISA on your PC. By default, NI-VISA is installed under C:\Program Files\IVI Foundation\VISA.



By now, VISA library has been added.

4. Add the Text, Edit, and Button controls. The layout interface for adding controls is as follows:



5. Add the control variables.

Click **View → ClassWizard**, and then click on the "Member Variables" tab to add the following three variables:
Instrument address: CString m_strInstrAddr
Command: CString m_strCommand
Returned value: CString m_strResult

6. Encapsulate the read and write operations of VISA.

1) Encapsulate the write operation of VISA for easier operation.

```
bool CDemoForRSADlg::InstrWrite(CString strAddr, CString strContent) //Write operation
{
 ViSession defaultRM,instr;
 ViStatus status;
 ViUInt32 retCount;
 char * SendBuf = NULL;
 char * SendAddr = NULL;
 bool bWriteOK = false;
 CString str;
```

```
    // Change the address's data style from CString to char*
    SendAddr = strAddr.GetBuffer(strAddr.GetLength());
    strcpy(SendAddr,strAddr);
    strAddr.ReleaseBuffer();

    // Change the command's data style from CString to char*
    SendBuf = strContent.GetBuffer(strContent.GetLength());
    strcpy(SendBuf,strContent);
    strContent.ReleaseBuffer();

    //Open a VISA resource
    status = viOpenDefaultRM(&defaultRM);
    if (status < VI_SUCCESS)
    {
        AfxMessageBox("No VISA resource was opened!");
        return false;
     }

    status = viOpen(defaultRM, SendAddr, VI_NULL, VI_NULL, &instr);

    //Write command to the instrument
    status = viWrite(instr, (unsigned char *)SendBuf, strlen(SendBuf), &retCount);

    //Close the system
    status = viClose(instr);
    status = viClose(defaultRM);

    return bWriteOK;
  }
```

2) Encapsulate the read operation of VISA for easier operation.

```
  bool CDemoForRSADlg::InstrRead(CString strAddr, CString *pstrResult) //Read operation
  {
    ViSession defaultRM,instr;
    ViStatus status;
    ViUInt32 retCount;
    char * SendAddr = NULL;
    unsigned char RecBuf[MAX_REC_SIZE];
    bool bReadOK = false;
    CString str;

    // Change the address's data style from CString to char*
    SendAddr = strAddr.GetBuffer(strAddr.GetLength());
    strcpy(SendAddr,strAddr);
    strAddr.ReleaseBuffer();

    memset(RecBuf,0,MAX_REC_SIZE);

    //Open a VISA resource
    status = viOpenDefaultRM(&defaultRM);
    if (status < VI_SUCCESS)
    {
        // Error Initializing VISA...exiting
        AfxMessageBox("No VISA resource was opened!");
        return false;
     }

    //Open the instrument
```

```
    status = viOpen(defaultRM, SendAddr, VI_NULL, VI_NULL, &instr);

    //Read from the instrument
    status = viRead(instr, RecBuf, MAX_REC_SIZE, &retCount);

    //close the system
    status = viClose(instr);
    status = viClose(defaultRM);

    (*pstrResult).Format("%s",RecBuf);

    return bReadOK;
}
```

3) Encapsulate the read operation with exception handling function of VISA.

```
ViStatus CDemoForRSADlg::OpenVisaDevice(CString strAddr)      //Open a VISA device
{
    ViStatus status;
    char * SendAddr = NULL;

    // Change the address's data style from CString to char*
    SendAddr = strAddr.GetBuffer(strAddr.GetLength());
    strcpy(SendAddr,strAddr);
    strAddr.ReleaseBuffer();

    //Open a VISA resource
    status = viOpenDefaultRM(&m_SessRM);

    if (status == 0)
    {
        //Open the device
        status = viOpen(m_SessRM, SendAddr, VI_NULL, VI_NULL, &m_SessInstr);

        //If you fails to open the connection, close the resource
        if (status != 0)
        {
            viClose(m_SessRM);
        }
    }

    return status;
}


ViStatus CDemoForRSADlg::CloseVisaDevice()        //Close a VISA device
{
    ViStatus status;

    //Close the device
    status = viClose(m_SessInstr);

    if (status == 0)
    {
        //close the resource
        status = viClose(m_SessRM);
    }

    return status;
}
```

```
bool CDemoForRSADlg::InstrWriteEx(CString strAddr, CString strContent) //Write operation with
exception handling
{
ViStatus status;
ViUInt32 retCount;
char * SendBuf = NULL;
bool bWriteOK = true;

// Change the address's data style from CString to char*
SendBuf = strContent.GetBuffer(strContent.GetLength());
strcpy(SendBuf,strContent);
strContent.ReleaseBuffer();

do
{
      //Write command to the instrument
      status = viWrite(m_SessInstr, (unsigned char *)SendBuf, strlen(SendBuf), &retCount);

      //If an error occurs, perform error handing
      if (status < 0)
      {
            //If the time exceed the limit value, resend the command after a delay of 1s
            if (VI_ERROR_TMO == status)
            {
                  Sleep(1000);
                   status = viWrite(m_SessInstr, (unsigned char *)SendBuf, strlen(SendBuf), &retCount);
            }
            else
            {
                  //If another error occurs, reopen the connection after the connection is closed and
                  resend the command
                  status = CloseVisaDevice();
                  Sleep(1000);
                  status = OpenVisaDevice(m_strInstrAddr);
                  if (status == 0)
                  {
                        status = viWrite(m_SessInstr, (unsigned char *)SendBuf, strlen(SendBuf),
                        &retCount);
                   }
             }
       }
} while (status < 0);

return bWriteOK;
}


bool CDemoForRSADlg::InstrReadEx(CString strAddr, CString *pstrResult) //Read operation with
exception handling
{
ViStatus status;
ViUInt32 retCount;
char * SendAddr = NULL;
unsigned char RecBuf[MAX_REC_SIZE];
bool bReadOK = true;

// Change the address's data style from CString to char*
SendAddr = strAddr.GetBuffer(strAddr.GetLength());
strcpy(SendAddr,strAddr);
```

```
strAddr.ReleaseBuffer();

memset(RecBuf,0,MAX_REC_SIZE);

do
{
    //Read from the instrument
    status = viRead(m_SessInstr, RecBuf, MAX_REC_SIZE, &retCount);
    if (status < 0)
    {
        //If the time exceeds the limit value, read from the instrument after a delay of 1s
        if (VI_ERROR_TMO == status)
        {
            Sleep(1000);
            status = viRead(m_SessInstr, RecBuf, MAX_REC_SIZE, &retCount);
        }
        else
        {
            //If another error occurs, reopen the connection after the connection is closed and
            reread from instrument
            status = CloseVisaDevice();
            Sleep(1000);
            status = OpenVisaDevice(m_strInstrAddr);
            if (status == 0)
            {
                status = viRead(m_SessInstr, RecBuf, MAX_REC_SIZE, &retCount);
            }
        }
    }

} while (status < 0);


(*pstrResult).Format("%s",RecBuf);

return bReadOK;
}
```

7. Add the control message response codes.

1) Connect to the instrument

```
void CDemoForRSADlg::OnBtConnectInstr()                // Connect to the instrument
{
//TODO: Add your control notification handler code here
ViStatus status;
ViSession defaultRM;
ViString expr = "?*";
ViPFindList findList = new unsigned long;
ViPUInt32 retcnt = new unsigned long;
ViChar instrDesc[1000];
CString strSrc = "";
CString strInstr = "";
unsigned long i = 0;
bool bFindRSA = false;

status = viOpenDefaultRM(&defaultRM);
if (status < VI_SUCCESS)
{
// Error Initializing VISA...exiting
MessageBox("No VISA instrument was opened ! ");
```

```
        return ;
        }

        memset(instrDesc,0,1000);

        // Find resource
        status = viFindRsrc(defaultRM,expr,findList, retcnt, instrDesc);

        for (i = 0;i < (*retcnt);i++)
        {
            // Get instrument name
            strSrc.Format("%s",instrDesc);
            InstrWrite(strSrc,"*IDN?");
            ::Sleep(200);
            InstrRead(strSrc,&strInstr);

          // If the instrument(resource) belongs to the RSA series then jump out //from the loop
             strInstr.MakeUpper();
             if (strInstr.Find("RSA") >= 0)
             {
                 bFindRSA = true;
                 m_strInstrAddr = strSrc;
                 break;
             }

            //Find next instrument
            status = viFindNext(*findList,instrDesc);
        }

        if (bFindRSA == false)
        {
           MessageBox("Didn't find any RSA!");
        }
        UpdateData(false);
    }
```

2) Write Operation

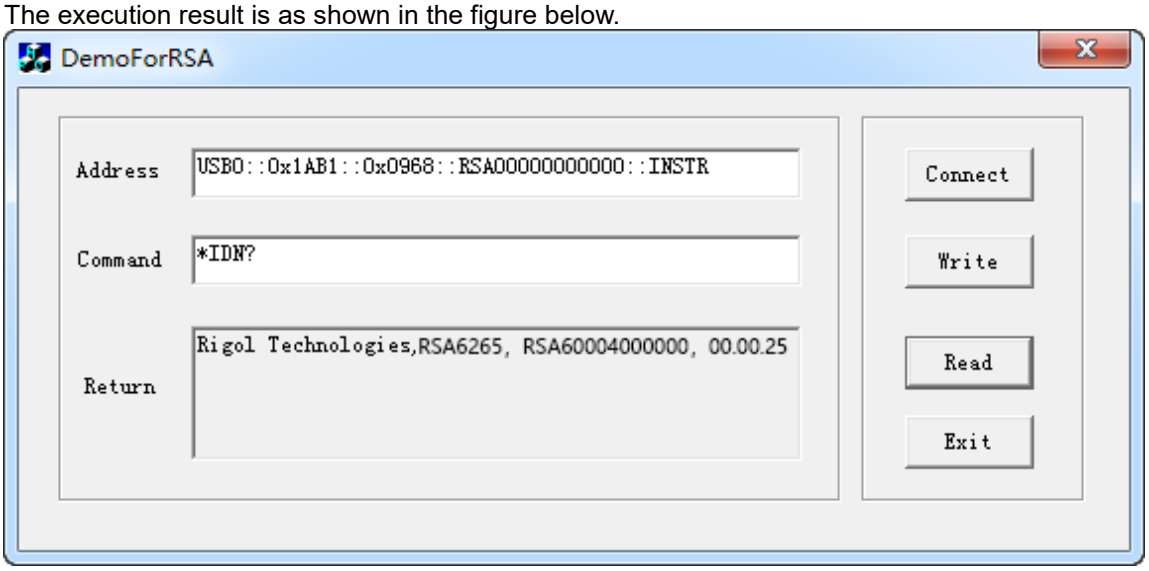
```
    void CDemoForRSADlg::OnBtWrite()                    //Write operation
    {
     //TODO: Add your control notification handler code here
     UpdateData(true);
     if (m_strInstrAddr.IsEmpty())
     {
         MessageBox("Please connect to the instrument first!");
      }
     InstrWrite(m_strInstrAddr,m_strCommand);
     m_strResult.Empty();
     UpdateData(false);
    }
```

3) Read Operation

```
    void CDemoForRSADlg::OnBtRead()                     //Read operation
    {
     //TODO: Add your control notification handler code here
     UpdateData(true);
     InstrRead(m_strInstrAddr,&m_strResult);
     UpdateData(false);
    }
```

8. Run the results.
   1) Click **Connect** to search for the spectrum analyzer;
   2) Input "*IDN?" in the "Command" edit box;
   3) Click **Write** to write the command into the spectrum analyzer;
   4) Click **Read** to read the return value.

The execution result is as shown in the figure below.

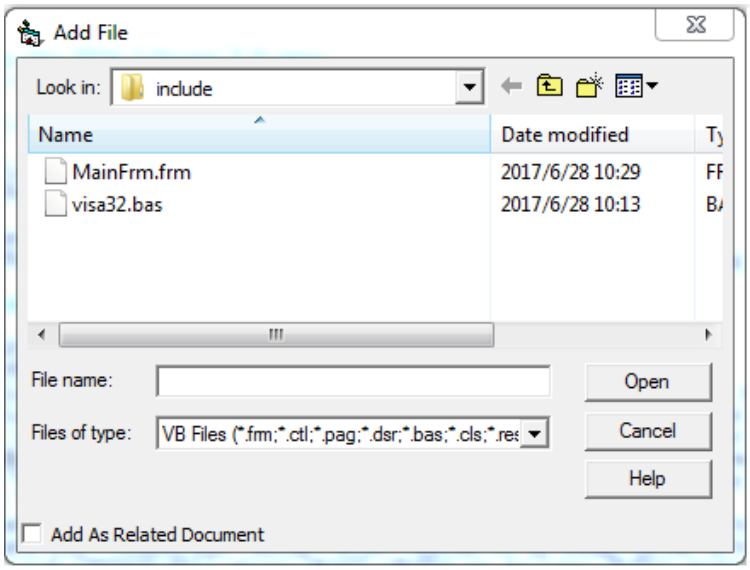


Visual Basic 6.0 Programming Example

Enter the Visual Basic 6.0 programming environment, and perform the following procedures.

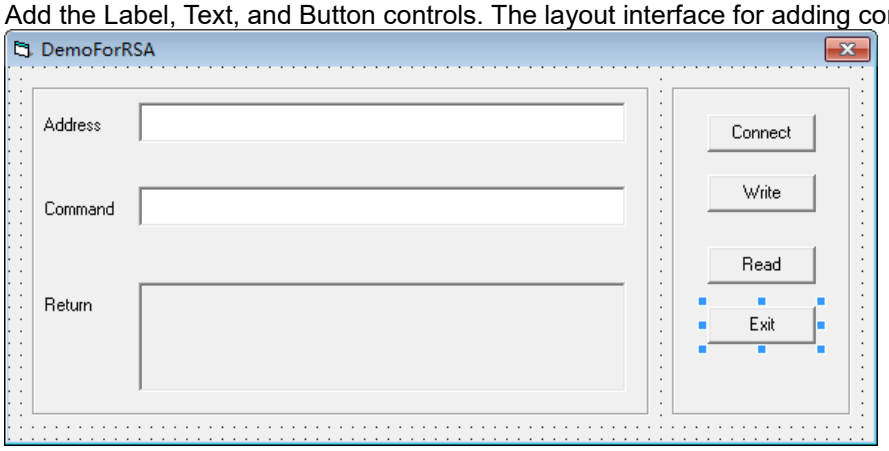
1. Build a standard application program project (Standard EXE), and name it "DemoForRSA".

2. Open **Project → Add File…**. Search for the visa32.bas file from the include folder in the installation path of NI-VISA, and then add the file to the project. The visa32.bas module contains all VISA functions and constant statements.



Then, add the Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long) statement into the visa32.bas module; or you can also create a new module to declare the Sleep function.

3. Add the Label, Text, and Button controls. The layout interface for adding controls is as follows:



4. Encapsulate the read and write operations of VISA.
   1) Encapsulate the write operation of VISA for easier operation.

```
'------------------------------------------------------------
'Function Name：InstrWrite
'Function：Send command to the instrument
'Input：rsrcName,instrument(resource) name
        strCmd,Command
'------------------------------------------------------------
Public Sub InstrWrite(rsrcName As String, strCmd As String)
    Dim status As Long
    Dim dfltRM As Long
    Dim sesn As Long
```

```
    Dim rSize As Long

    'Initialize the system
    status = viOpenDefaultRM(dfltRM)
    'Failed to initialize the system
    If (status < VI_SUCCESS) Then
        MsgBox " No VISA resource was opened！ "
        Exit Sub
    End If
    'Open the VISA instrument
    status = viOpen(dfltRM, rsrcName, VI_NULL, VI_NULL, sesn)
    'Failed to open the instrument
    If (status < VI_SUCCESS) Then
        MsgBox "Failed to open the instrument！ "
        Exit Sub
    End If

    'Write command to the instrument
    status = viWrite(sesn, strCmd, Len(strCmd), rSize)
    'Failed to write to the instrument
    If (status < VI_SUCCESS) Then
        MsgBox " Faild to write to the instrument！ "
        Exit Sub
    End If

    'Close the system
    status = viClose(sesn)
    status = viClose(dfltRM)

End Sub
```

2) Encapsulate the read operation of VISA for easier operation.

```
'------------------------------------------------------------
'Function Name：InstrRead
'Function：Read the return value from the instrument
'Input：rsrcName,Resource name
'Return：The string gotten from the instrument
'------------------------------------------------------------
Public Function InstrRead(rsrcName As String) As String
    Dim status As Long
    Dim dfltRM As Long
    Dim sesn As Long
    Dim strTemp0 As String * 256
    Dim strTemp1 As String
    Dim rSize As Long

    'Begin by initializing the system
    status = viOpenDefaultRM(dfltRM)
    'Initialization failed
    If (status < VI_SUCCESS) Then
        MsgBox " Failed to open the instrument! "
        Exit Function
    End If
    'Open the instrument
    status = viOpen(dfltRM, rsrcName, VI_NULL, VI_NULL, sesn)
    'Failed to open the instrument
    If (status < VI_SUCCESS) Then
        MsgBox " Failed to open the instrument! "
        Exit Function
```

```
    End If

    'Read from the instrument
    stasus = viRead(sesn, strTemp0, 256, rSize)
    'Reading failed
    If (status < VI_SUCCESS) Then
        MsgBox " Failed to read from the instrument! "
        Exit Function
    End If

    'Close the system
    status = viClose(sesn)
    status = viClose(dfltRM)

    'Remove the space at the end of the string
    strTemp1 = Left(strTemp0, rSize)
    InstrRead = strTemp1
End Function
```

5. Add the control event codes.

1) Connect to the instrument

```
'Connect to the instrument
Private Sub CmdConnect_Click()
    Const MAX_CNT = 200
    Dim status As Long
    Dim dfltRM As Long
    Dim sesn As Long
    Dim fList As Long
    Dim buffer As String * MAX_CNT, Desc As String * 256
    Dim nList As Long, retCount As Long
    Dim rsrcName(19) As String * VI_FIND_BUFLEN, instrDesc As String * VI_FIND_BUFLEN
    Dim i, j As Long
    Dim strRet As String
    Dim bFindRSA As Boolean

    'Initialize the system
    status = viOpenDefaultRM(dfltRM)
    'Initialization failed
    If (status < VI_SUCCESS) Then
        MsgBox " No VISA resource was opened  ! "
        Exit Sub
    End If

    'Find instrument resource
    Call viFindRsrc(dfltRM, "USB?*INSTR", fList, nList, rsrcName(0))
    'Get the list of the instruments (resources)
    strRet = ""
    bFindRSA = False
    For i = 0 To nList - 1
    'Get the instrument name
    InstrWrite rsrcName(i), "*IDN?"
    Sleep 200
    strRet = InstrRead(rsrcName(i))
    'Continuing searching for the resource until an RSA instrument is found
    strRet = UCase(strRet)
    j = InStr(strRet, "RSA")
    If (j >= 0) Then
        bFindRSA = True
        Exit For
```

```
    End If

    Call viFindNext(fList + i - 1, rsrcName(i))
    Next i
    'Display
    If (bFindRSA = True) Then
       TxtInsAddr.Text = rsrcName(i)
    Else
       TxtInsAddr.Text = ""
    End If
End Sub
```

2) Write Operation

```
'Write the command to the instrument
Private Sub CmdWrite_Click()
   If (TxtInsAddr.Text = "") Then
       MsgBox ("Please write the instrument address！ ")
   End If

   InstrWrite TxtInsAddr.Text, TxtCommand.Text
End Sub
```

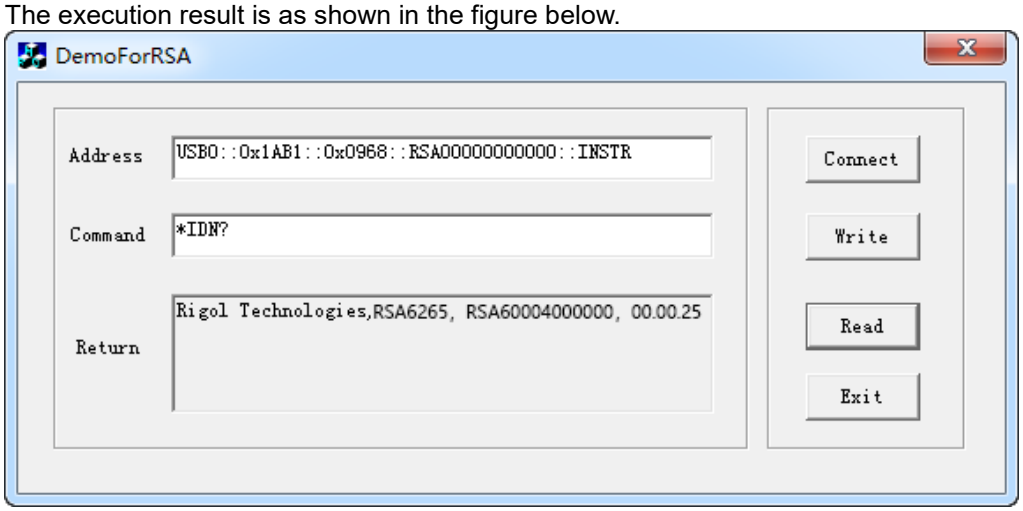
3) Read Operation

```
'Read the return value from the instrument
Private Sub CmdRead_Click()
   Dim strTemp As String
   strTemp = InstrRead(TxtInsAddr.Text)
   TxtReturn.Text = strTemp
End Sub
```

6. Run the results.

1) Click **Connect** to search for the spectrum analyzer;
2) Input "*IDN?" in the "Command" edit box;
3) Click **Write** to write the command into the spectrum analyzer;
4) Click **Read** to read the return value.

The execution result is as shown in the figure below.

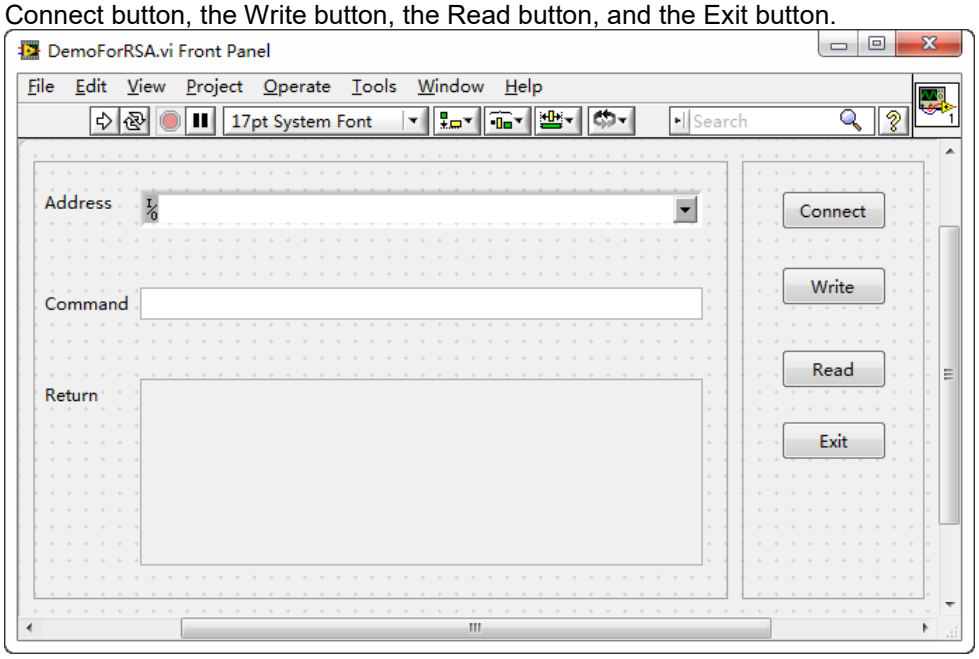


LabVIEW 2010 Programming Example

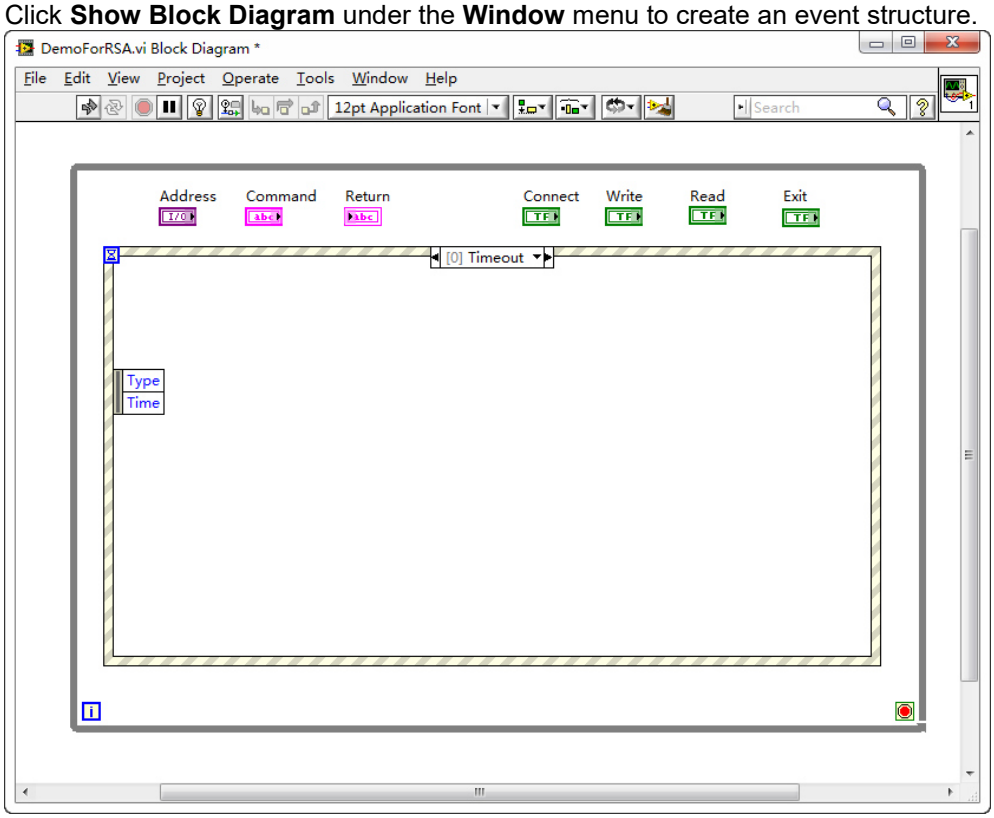
Enter the Labview 2010 programming environment, and perform the following procedures.

1. Create a VI file, and name it "DemoForRSA".

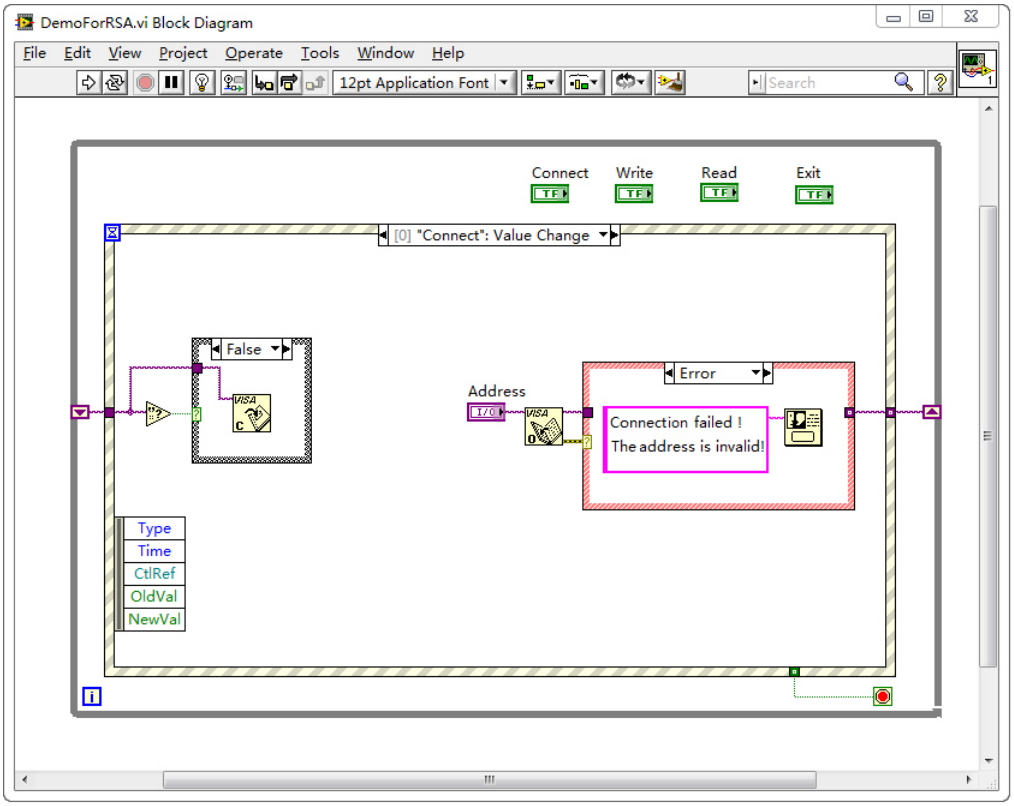
2. Add controls to the front panel interface, including the Address field, Command field, and Return field, the Connect button, the Write button, the Read button, and the Exit button.



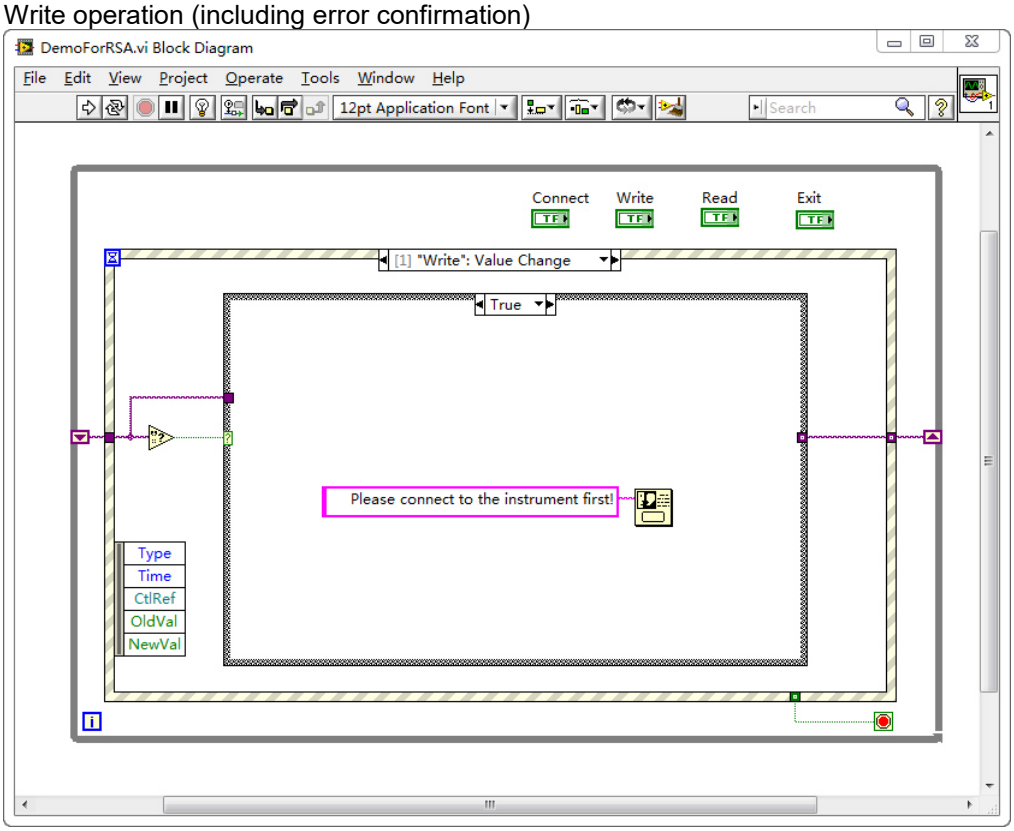
3. Click **Show Block Diagram** under the **Window** menu to create an event structure.

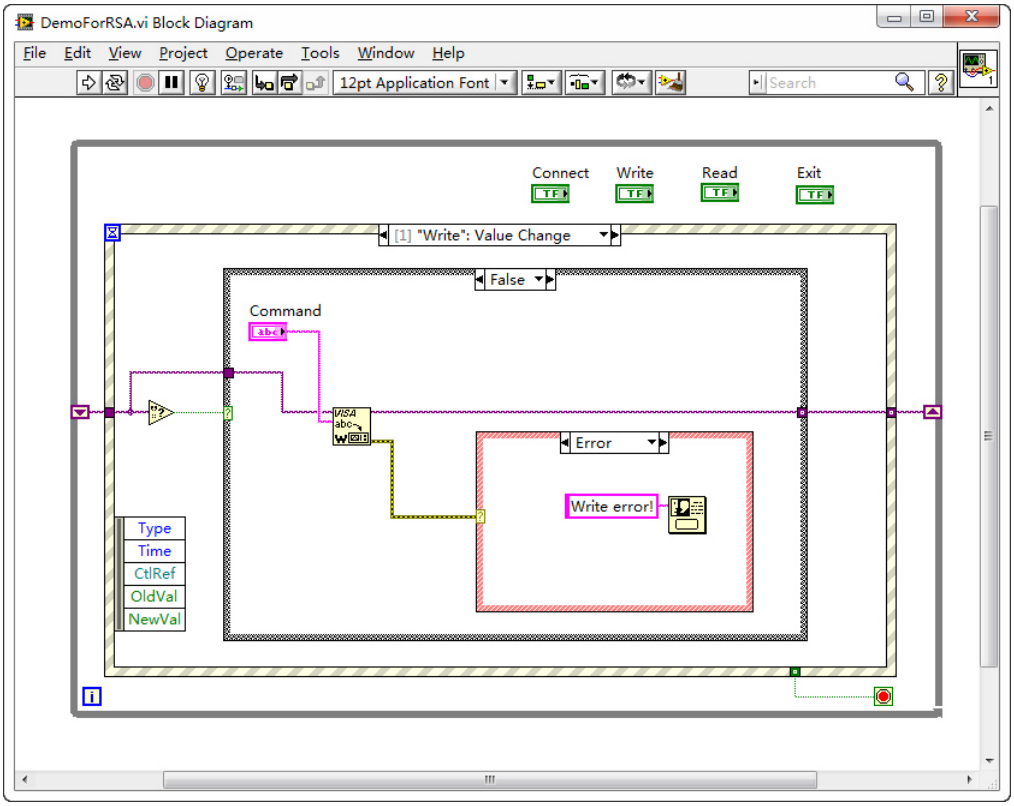


4. Add the events (including connecting to the instrument, write operation, read operation, and exit)
   1) Connect to the instrument

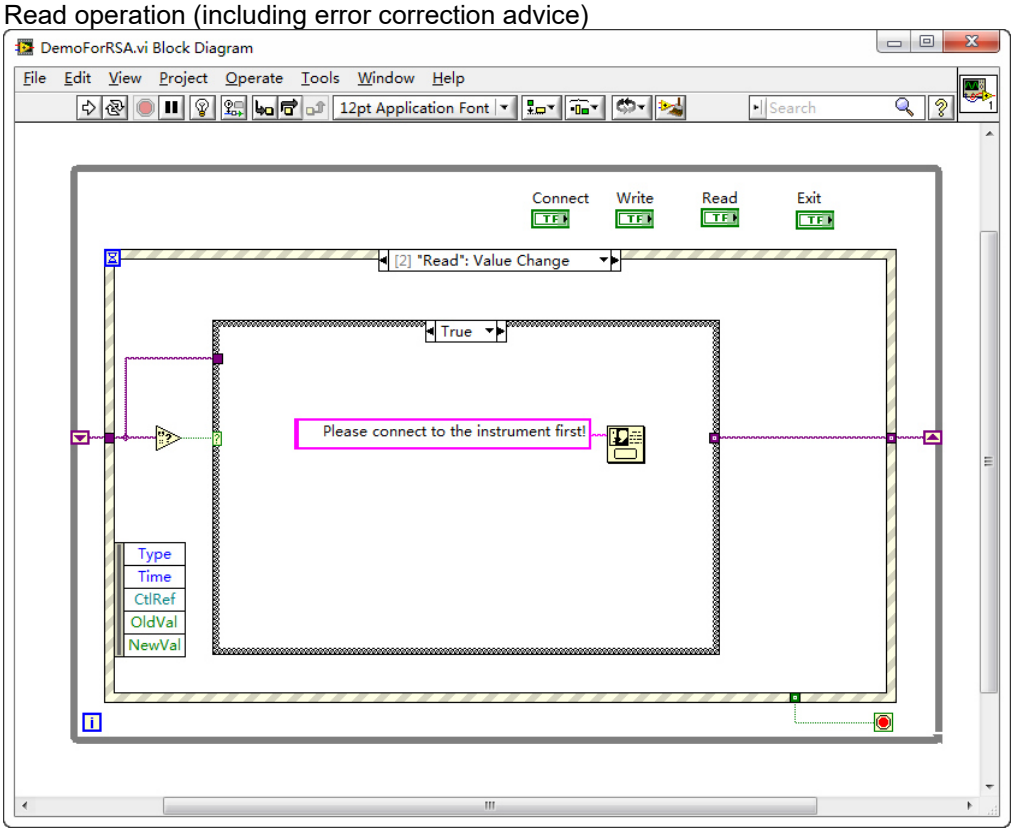


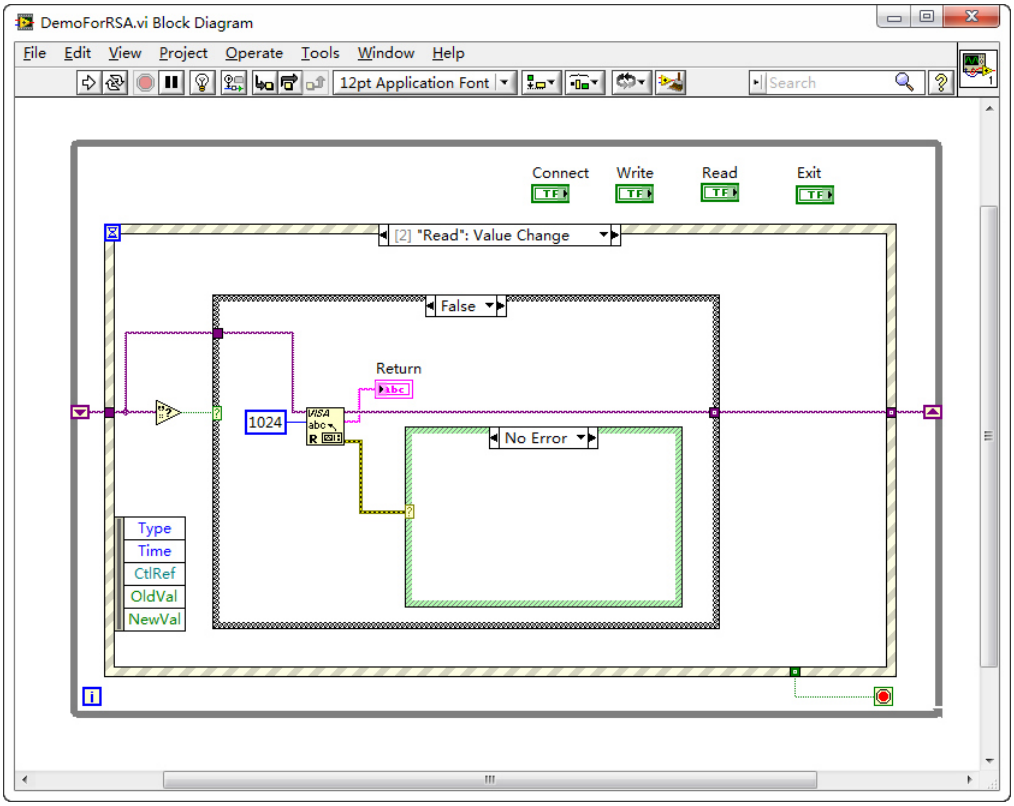
2) Write operation (including error confirmation)



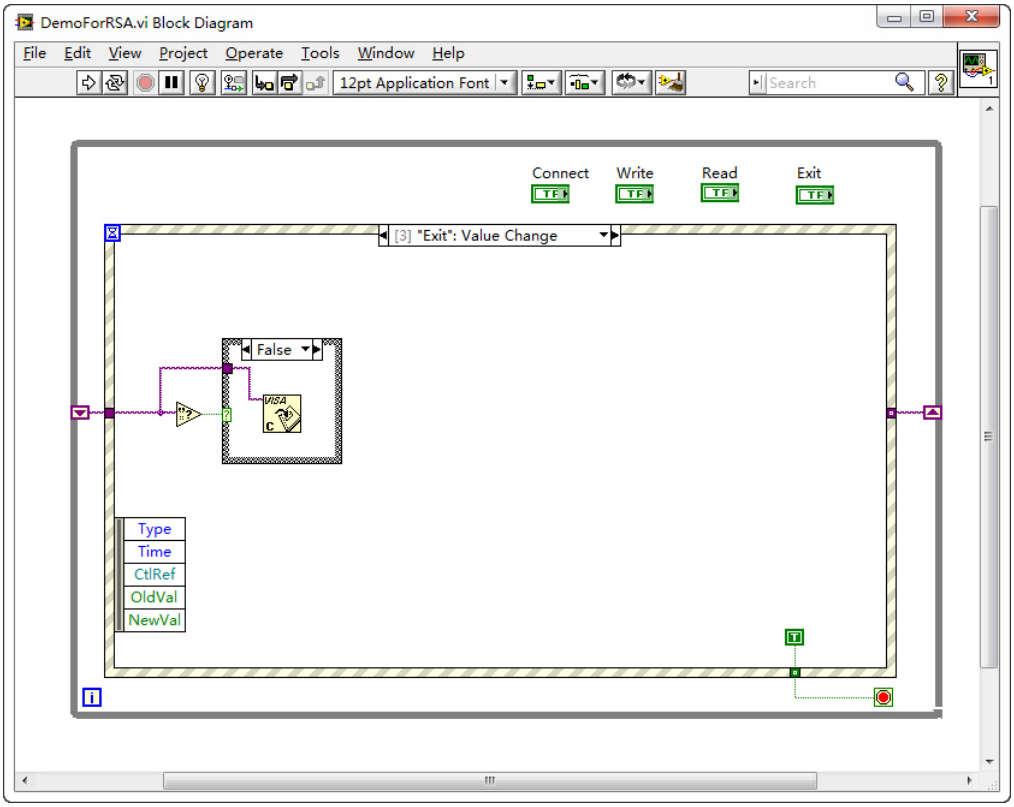


3) Read operation (including error correction advice)



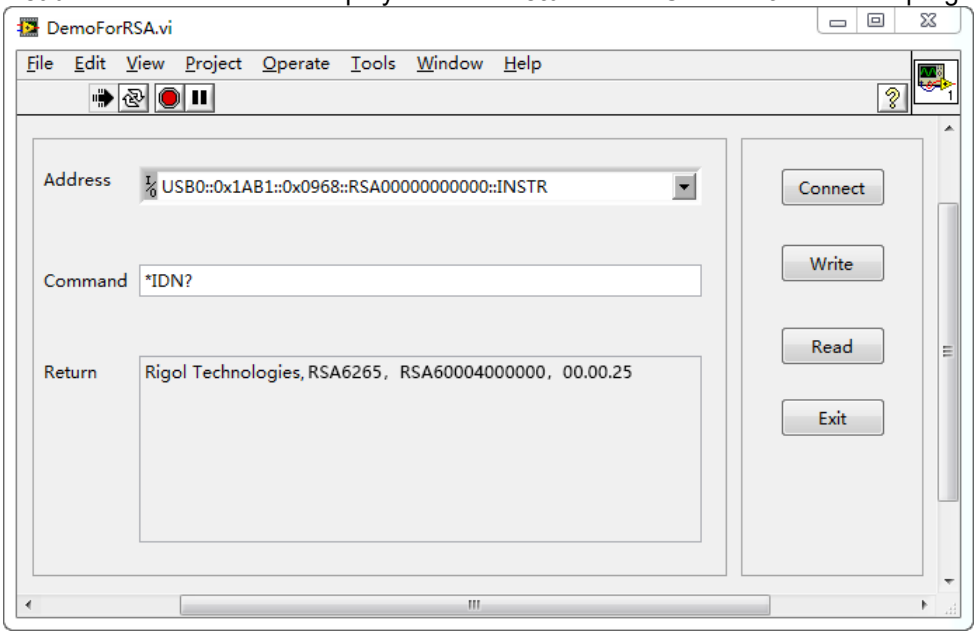


4) Exit



5) Run the program, and then the following interface is displayed below. Click the VISA resource name from the drop-down list under **Address**, and click **Connect** to connect the instrument. Then, input a command in the **Command** field. Click **Write** to write the command to the instrument. If the command is a query (e.g.*IDN?), click **Write** to write the command into the instrument, and then click

Read. The return value is displayed in the **Return** field. Click **Exit** to exit the program.



Linux Programming Example

This section illustrates how to program and control the spectrum analyzer to realize the common functions in Linux operating system.

Programming Preparations

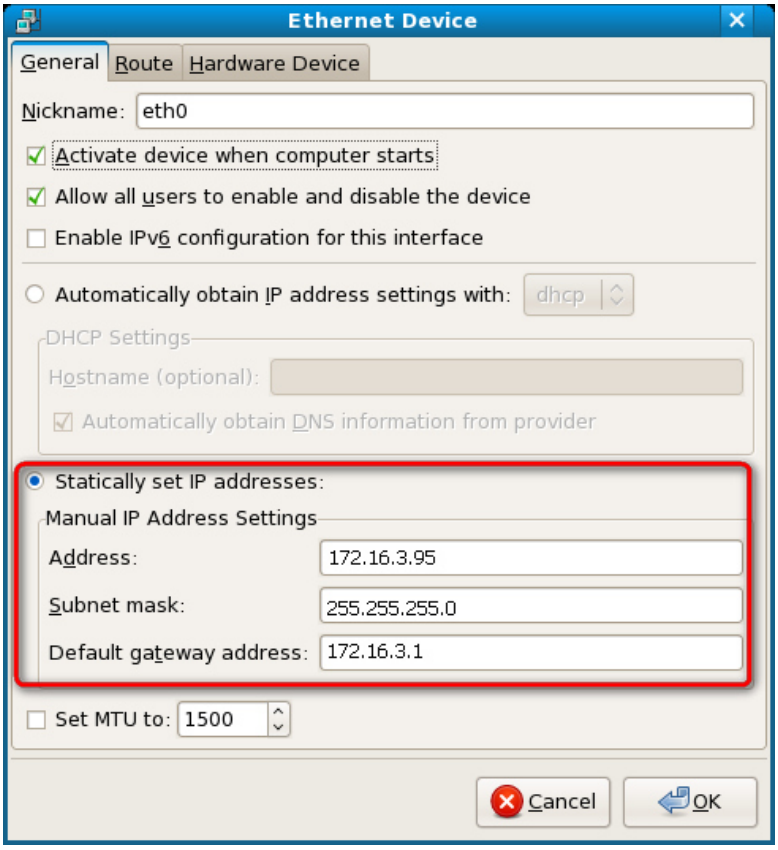
1. Programming environment:
   Operating system: Fedroa 8 (Linux-2.6.23)
   GCC version: gcc-4.1.2

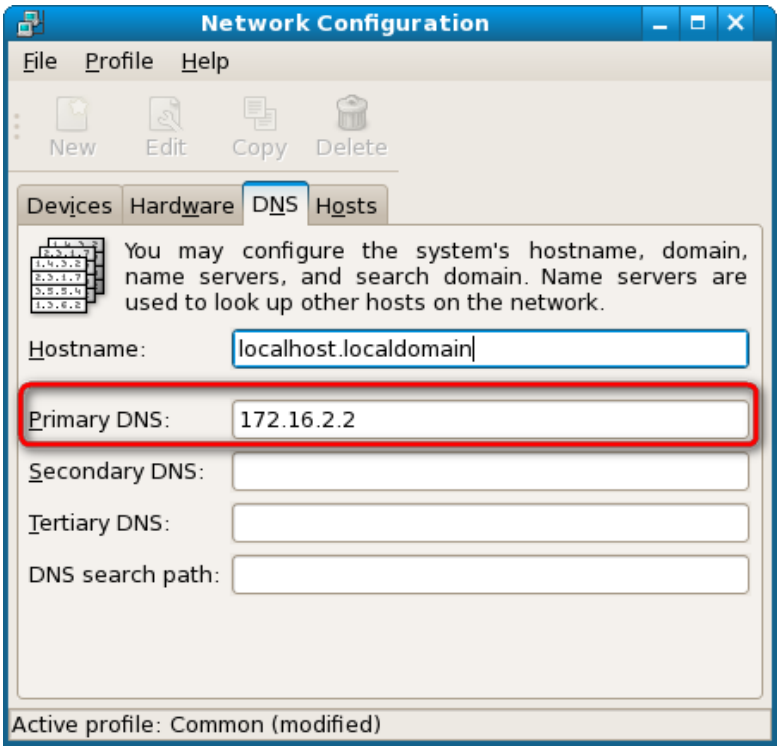
2. Install the VISA library. First, check whether your PC has installed NI's VISA library. If not, download it from NI website (http://www.ni.com/visa/). The installation procedures are as follows:
   Download the VISA library NI-VISA-4.4.0.ISO from the NI website.

   Create a new directory.
   #mkdir NI_VISA
   Mount the iso file.
   #mount -o loop -t iso9660 NI-VISA-4.4.0.iso NI_VISA
   Enter the NI_VISA directory to install
   #cd NI_VISA
   #./INSTALL
   Unmount the iso file
   #umount NI_VISA
   After the installation is finished, the default installation path is /usr/local.

3. Connect spectrum analyzer to the PC via the LAN interface of the analyzer. Use the USB cable to connect the analyzer to the PC via the LAN interface on the rear panel of the analyzer. You can also use a network cable to connect the spectrum analyzer to the local area network where the PC resides.

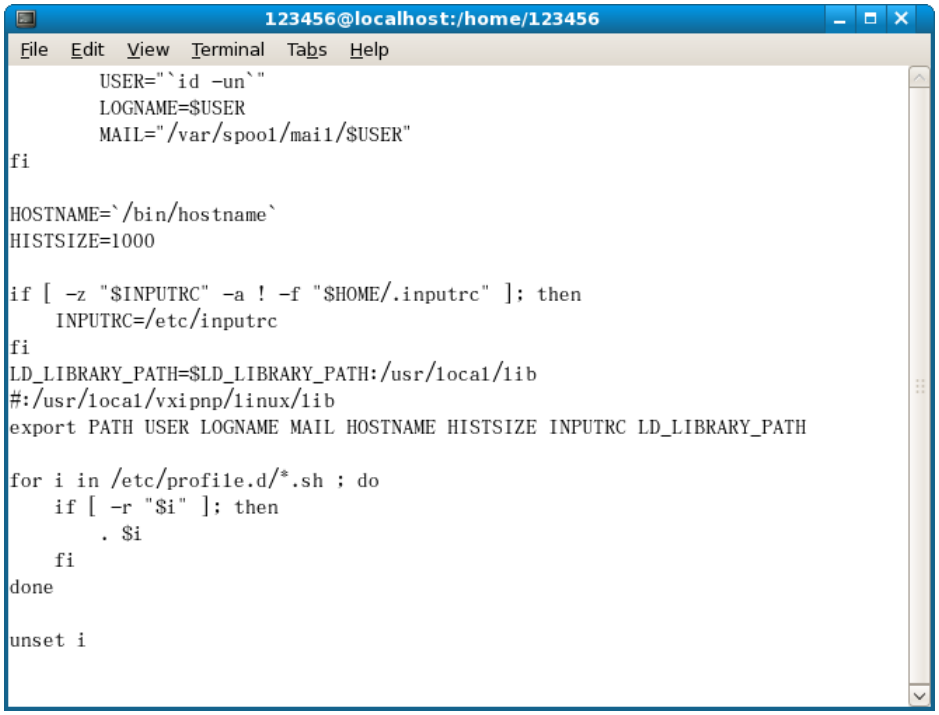
   After the spectrum analyzer is connected to the PC properly, configure the network address for the spectrum analyzer to make its address to be within the same network segment where the PC resides. For example, if the network address and DNS setting configured for the PC are as shown in the figures below, then, the network address of the spectrum analyzer should be configured as follows.
   IP Address: 172.16.3.X*
   Default Gateway: 172.16.3.1
   Subnet Mask: 255.255.255.0
   DNS: 172.16.2.2
   Note*: X can be any value ranging from 2 to 254.



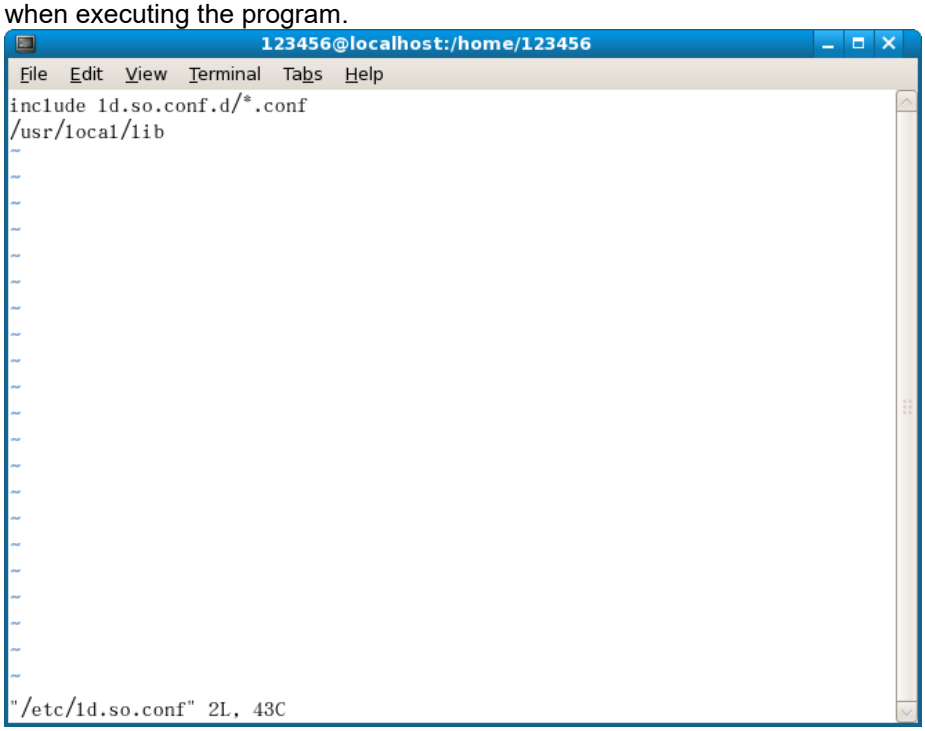


4. Use either of the following two methods to add the library location to the search path of the library, so that the program can load the installed library file automatically.

   Method 1: Specify the search path of the library in the environment variable LD_LIBRARY_PATH.
   Operation Method: Add the library file path /usr/local/lib to the LD_LIBRARY_PATH variable in the /etc/profile file, as shown in the figure below.



Method 2: Add the search path of the library in the /etc/ld.so.conf file.
Operation Method: #echo "/usr/local/lib" >> /etc/ld.so.conf, as shown in the figure below.
After setting the search path of the library in /etc/ld.so.conf, run the /sbin/ldconfig command to update /etc/ld.so.cache (this command should have the root permission) to ensure the location of the library when executing the program.



Linux Programming Procedures

1. Edit the DemoForDSA.h header file and declare a class to encapsulate the operation and property of the instrument.

```
#ifndef DEMO_FOR_RSA_H
#define DEMO_FOR_RSA_H

#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <iostream>
//#include <syswait.h>
using namespace std;

#define MAX_SEND_BUF_SIZE 50
#define MAX_REC_SIZE 300

class DemoForRSA
{
// Construction
public:
DemoForRSA();
bool InstrRead(string strAddr, string & pstrResult);
bool InstrWrite(string strAddr, string strContent);
bool ConnectInstr();

string m_strInstrAddr;
string m_strResult;
string m_strCommand;

};

void makeupper(string & instr);

#endif
```

2. Edit the DemoForDSA.cpp file to realize various operations of the instrument.

```
#include "visa.h"
#include "DemoForRSA.h"

DemoForRSA::DemoForRSA()
{
m_strInstrAddr = "";
m_strResult = "";
m_strCommand = "";
}

bool DemoForRSA::ConnectInstr()

{
ViUInt32 retCount;
ViStatus status;
ViSession defaultRM;
ViString expr = "?*";
ViPFindList findList = new unsigned long;
ViPUInt32 retcnt = new unsigned long;
string strSrc = "";
string strInstr = "";
ViChar instrDesc[1000];

unsigned long i = 0;
bool bFindRSA = false;
memset(instrDesc,0,1000);
```

```
//Turn on the VISA device
status = viOpenDefaultRM(&defaultRM);

if (status < VI_SUCCESS)
{
   cout<<"No VISA equipment!"<<endl;
   return false;
}

//Search for resources
status = viFindRsrc(defaultRM,expr,findList, retcnt, instrDesc);

for (i = 0;i < (*retcnt);i++)
{
   //Acquire the instrument name
   strSrc = instrDesc;

   InstrWrite(strSrc,"*IDN?");
   usleep(200);
   InstrRead(strSrc,strInstr);

   // If the RSA series is found, then exit
   makeupper(strInstr);
   if (strInstr.find("RSA",0) > 0)
   {
      bFindRSA = true;
      m_strInstrAddr = strSrc;
      break;
    }

   //Acquire the next device
   status = viFindNext(*findList,instrDesc);
}

if (bFindRSA == false)
{
    printf("RSA device not found!\n");
    return false;
}

    return true;
}

bool DemoForRSA::InstrWrite(string strAddr, string strContent) //Write operation
{
ViSession defaultRM,instr;
ViStatus status;
ViUInt32 retCount;
char * SendBuf = NULL;
char * SendAddr = NULL;
bool bWriteOK = false;
string str;
//Address conversion, convert the string type to char*
SendAddr = const_cast<char*>(strAddr.c_str());

//Address conversion, convert the string type to char*
SendBuf = const_cast<char*>(strContent.c_str());
```

```
//Turn on the specified device
status = viOpenDefaultRM(&defaultRM);
if (status < VI_SUCCESS)
{
   cout<<"No VISA equipment!"<<endl;
   return false;
}

status = viOpen(defaultRM, SendAddr, VI_NULL, VI_NULL, &instr);

//Write command to the device
status = viWrite(instr, (unsigned char *)SendBuf, strlen(SendBuf), &retCount);

//Turn off the device
status = viClose(instr);
status = viClose(defaultRM);
return bWriteOK;
}

bool DemoForRSA::InstrRead(string strAddr, string & pstrResult) //Read operation
{
ViSession defaultRM,instr;
ViStatus status;
ViUInt32 retCount;
char* SendAddr = NULL;
char * result = NULL;
bool bReadOK = false;
unsigned char RecBuf[MAX_REC_SIZE];
string str;
memset(RecBuf,0,MAX_REC_SIZE);

result=char*)malloc(MAX_REC_SIZE*sizeof(char));
memset(result,0,MAX_REC_SIZE);

//Address conversion, convert the string type to char*
SendAddr=const_cast<char*>(strAddr.c_str());

//Turn on the VISA device
status = viOpenDefaultRM(&defaultRM);
if (status < VI_SUCCESS)
{
    // Error Initializing VISA...exiting
    cout<<"No VISA equipment!"<<endl;
    return false;
}

//Turn on the specified device
status = viOpen(defaultRM, SendAddr, VI_NULL, VI_NULL, &instr);

//Read from the device
status = viRead(instr, RecBuf, MAX_REC_SIZE, &retCount);

//Turn off the device
status = viClose(instr);
status = viClose(defaultRM);
sprintf(result,"%s",RecBuf);
pstrResult = result;
free(result);
return bReadOK;
```

```
}

void makeupper( string &instr)
{

   string outstr = "";
   if(instr == "")
   {
      exit(0);
   }

   for(int i = 0;i < instr.length();i++)
   {
      instr[i] = toupper(instr[i]);
   }

}
```

3. Edit the function file mainloop.cpp to complete the flow control.

```
#include "DemoForRSA.h"

void menudisplay()
{
   cout<<"\t\t   Please operate the instrument:\n read write quit"<<endl;
}

int main()
{
   DemoForRSA demo;
   char temp[50];
   if(!demo.ConnectInstr())
   {
      cout<<"can not connect the equipment!"<<endl;
      return 0;
    }
   else
   {

      cout<<"\n connect equipment success!"<<endl;
      cout<<" the equipment address is :"<<demo.m_strInstrAddr<<endl;
    }

   while(1)
   {
      menudisplay();
      //cin>>demo.m_strCommand;
      cin.getline(temp,50);
      demo.m_strCommand=temp;
      if(demo.m_strCommand[0]='r' && demo.m_strCommand[1]='e'
         && demo.m_strCommand[2]='a' && demo.m_strCommand[3]='d')
      {
         //demo.InstrWrite(demo.m_strInstrAddr,"*IDN?");
         //demo.InstrRead(demo.m_strInstrAddr,demo.m_strResult);
         cout<<"read result:"<<demo.m_strResult<<endl;
         demo.m_strResult="";

       }

       else if (demo.m_strCommand[0]='w' && demo.m_strCommand[1]='r'
```

```
                && demo.m_strCommand[2]='i' && demo.m_strCommand[3]='t' &&
               demo.m_strCommand[4]='e')
        {
                if (demo.m_strInstrAddr="")
                {
                    cout<<"Please connect the instrument！ \n";
                 }
                demo.InstrWrite(demo.m_strInstrAddr,demo.m_strCommand.substr(5,40));
                usleep(200);

               //Read operation
              demo.InstrRead(demo.m_strInstrAddr,demo.m_strResult);

         }

        else if (demo.m_strCommand[0] == 'q' && demo.m_strCommand[1] == 'u'
                 && demo.m_strCommand[2] == 'i' && demo.m_strCommand[3] == 't')

        {
                break;
         }
        else if(demo.m_strCommand != "")
        {
                cout<<"Bad command!"<<endl;
         }
    }
    return 1;

}
```

4. makefile file

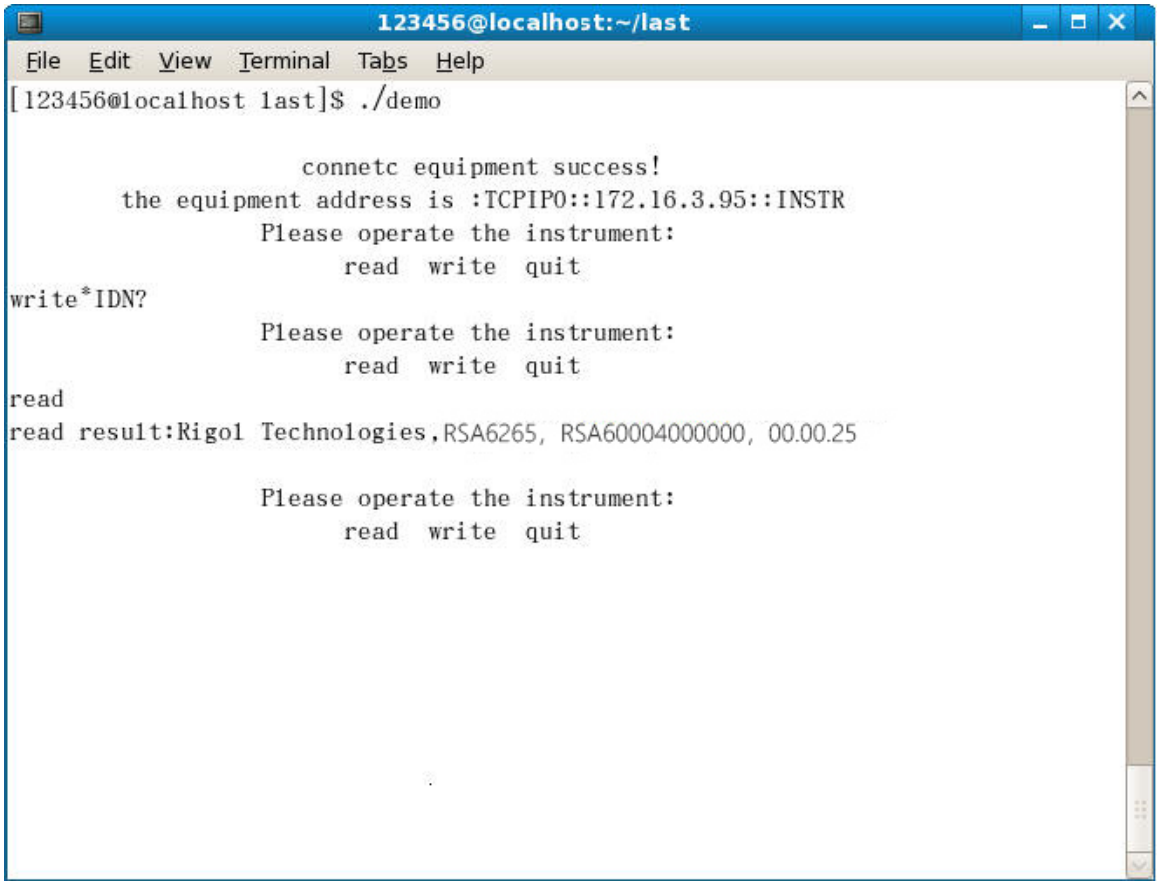
```
src = DemoForRSA.cpp mainloop.cpp DemoForRSA.h

obj = DemoForRSA.o mainloop.o
INCLUDE= -I/usr/local/vxipnp/linux/include
LIB= -lvisa -lc -lpthread
CC=
demo : $(obj)
$(CC) $(INCLUDE) $(LIB) -o demo $(obj)

mainloop.o : mainloop.cpp DemoForRSA.h
$(CC) -c $< -o $@
DemoForRSA.o: DemoForRSA.cpp DemoForRSA.h
$(CC) -c $< -o $@

.PHONY : clean
clean:
rm demo $(obj)
```

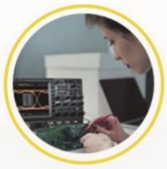
5. Run the results.
   1) #make
   2) ./demo
   3) When the program runs, the instrument is connected automatically. If no instrument is found, a prompt message "No VISA equipment!" is displayed, and the system exits the program. If the instrument is found and successfully connected, the following interface is displayed.
   4) Input write<command> (for example, write<*IDN?>) to write the command into the spectrum analyzer.
   5) Input read to read the return value, as shown in the figure below.



Boost Smart World and Technology Innovation

Industrial Intelligent Manufacturing

Semiconductors



Education& Research

Communication

System Integration

New Energy

- Cellular-5G/WIFI
- UWB/RFID/ ZIGBEE
- Digital Bus/Ethernet
- Optical Communication

- Digital/Analog/RF Chip
- Memory and MCU Chip
- Third-Generation Semiconductor
- Solar Photovoltaic Cells

- New Energy Automobile
- PV/Inverter
- Power Test
- Automotive Electronics

Provide Testing and Measuring Products and Solutions for Industry Customers

HEADQUARTER

RIGOL TECHNOLOGIES CO., LTD.
No.8 Keling Road, New District,
Suzhou,JiangSu, P.R.China
Tel: +86-400620002
Email: info-cn@rigol.com

EUROPE

RIGOL TECHNOLOGIES EU GmbH
Friedrichshafener Str. 5c
82205 Gilching
Germany
Tel: +49-(0)8105-27292-22
Email: johannes.kroiher@rigol.com

NORTH AMERICA

RIGOL TECHNOLOGIES, USA INC.
10220 SW Nimbus Ave.
Suite K-7
Portland, OR 97223
Tel: +1-877-4-**RIGOL**-1
Email: sales@rigol.com

JAPAN

RIGOL JAPAN CO., LTD.
5F,3-45-6,Minamiotsuka, Toshima-Ku,
Tokyo,170-0005,Japan
Tel: +81-3-6262-8932
Fax: +81-3-6262-8933
Email: info.jp@rigol.com

KOREA

RIGOL KOREA CO,. LTD.
5F, 222, Gonghang-daero,
Gangseo-gu, Seoul, Republic of Korea
Tel: +82-2-6953-4466
Fax: +82-2-6953-4422
Email: info.kr@rigol.com

For Assistance in Other Countries

Email: info.int@rigol.com