

# **RIGOL**

## **Programming Guide**

### **DSA1000 Series Spectrum Analyzer**

**May 2011**

**RIGOL Technologies, Inc.**



# Guaranty and Declaration

## Copyright

© 2010 RIGOL Technologies, Inc. All Rights Reserved.

## Trademark Information

**RIGOL** is a registered trademark of **RIGOL** Technologies, Inc.

## Publication Number

PGD02104-1110

## Notices

- **RIGOL** products are protected by patent law in and outside of P.R.C.
- **RIGOL** reserves the right to modify or change parts of or all the specifications and pricing policies at company's sole decision.
- Information in this publication replaces all previously corresponding material.
- **RIGOL** shall not be liable for losses caused by either incidental or consequential 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 copy or photocopy or rearrange without prior written approval of **RIGOL**.

## Product Certification

**RIGOL** guarantees this product conforms to the standards of national and industrial. Meanwhile, the related standards conform to other ISO will get further. At present, DSA1000 Series has passed CE, cTUVus, PictBridge and LXI certification.

## Contact Us

If you have any problem or requirement when using our products, please contact **RIGOL** Technologies, Inc. or your local distributors, or visit: [www.rigol.com](http://www.rigol.com)

## Contents of this Book

This manual is written for users who want to use remote commands to control **RIGOL** DSA1000 Series Spectrum Analyzer for measurement tasks. We believe that readers of this manual have read User's Guide for **RIGOL** DSA1000 Series Spectrum Analyzer carefully and gotten familiar with usage of the analyzer.

The manual contains the following topics:

- Chapter 1 summarizes the remote command programming and lists some relevant provisions of the SCPI commands.
- Chapter 2 introduces the command system of DSA1000 Series Spectrum Analyzer in details.
- Chapter 3 gives several application examples of the command system.
- The Appendix lists all DSA1000 Series Spectrum Analyzer commands from A to Z.

# Contents

<b>Guaranty and Declaration .....</b>	<b>I</b>
<b>Chapter 1 Overview.....</b>	<b>1-1</b>
Programming Introduction.....	1-2
SCPI Commands Introduction .....	1-3
Syntax .....	1-4
Symbol Description .....	1-5
Parameter Type .....	1-6
Command Abbreviation.....	1-7
<b>Chapter 2 Command Systems.....</b>	<b>2-1</b>
IEEE 488.2 .....	2-2
:ABORt.....	2-5
:CALCulate.....	2-6
:CALibration.....	2-25
:CONFigure.....	2-26
:COUPle .....	2-31
:DISPlay .....	2-32
:FETCh.....	2-38
:FORMat.....	2-46
:HCOpy .....	2-47
:INITiate .....	2-51
:INPut.....	2-54
:MMEMory .....	2-55
:OUTPut.....	2-61
:READ .....	2-62
[:SENSe] .....	2-70
:SOURce.....	2-99
:STATus.....	2-103
:SYSTem.....	2-106
:TRACe.....	2-114
:TRIGer.....	2-120
:UNIT.....	2-121
<b>Chapter 3 Programming Examples .....</b>	<b>3-1</b>
Prepare for Programming .....	3-2
Program in Visual C++ 6.0 .....	3-4
Program in Visual Basic 6.0 .....	3-11

Program in LabVIEW 8.6.....	3-17
Linux Programming Example .....	3-22
Prepare for Programming .....	3-22
Linux Programming Guide.....	3-25
<b>Command Quick Reference A-Z .....</b>	<b>1</b>

# Chapter 1 Overview

This chapter introduces you the remote command programming and some relevant provisions of the SCPI (Standard Commands for Programmable Instruments) commands, which mainly includes:

- Programming Introduction
- SCPI Commands Introduction
  - Syntax
  - Symbol Description
  - Parameter Type
- Command Abbreviation

## Programming Introduction

DSA1000 provides two kinds of interfaces for connecting with a computer: USB and GPIB. For the specified method of application, refer to the User's Guide of this product.

While you program using commands, all command words are transmitted and identified as ASCII string that enable users to control and carry out secondary development.

By programming, you can:

- Setup the analyzer.
- Make a measurement.
- Obtain data from the analyzer (such as the working status of the analyzer and the measurement results).

## SCPI Commands Introduction

SCPI (Standard Commands for Programmable Instrument) is based on IEEE 488.2 and usually divided into two sections: Common Commands and Control Commands defined for SCPI Instruments.

A common command is defined by IEEE 488.2 and must be executed as well as both the syntax and semantics of it follows the application of IEEE 488.2. The common command works independently of measurement and is used for controlling the reset, self-test and status operations. For more details, refer to "**IEEE 488.2**".

A Control Command defined for SCPI Instrument is used to measure and read data, control the state of a function or mode and so on, involving all measurement functions and some specific functional functions.

## Syntax

SCPI commands present a hierarchical tree structure and have more sub-systems. Each sub-system contains a root keyword and one or more sub-keywords. A command line usually begins with a ":" which is also used to separate different keywords; parameters are permitted to follow the keyword; "?" appeared following a command line denotes to query; "space" is used to separate the command and parameter.

For example:

```
:CALCulate:BANDwidth:NDB <rel_ampl>  
:CALCulate:BANDwidth:NDB?
```

**CALCulate** is the root keyword of the command above. **BANDwidth** and **NDB** are the second and third keyword separately. The command line begins with a ":" and uses ":" to separate each keyword; **<rel\_ampl>** denotes a assignable parameter; "?" denotes to query; command **":CALCulate:BANDwidth:NDB"** and parameter **<rel\_ampl>** are separated by a space.

"," is generally used for separating different parameters contained in the same command, such as:

```
:SYSTEM:DATE <year>,<month>,<day>
```

## Symbol Description

The following four symbols are not the content of SCPI commands but usually used in instances of the parameters from Explanation commands.

### 1. Braces { }

The parameters enclosed in braces are optional and can be ignored or set one or more times, such as:

```
[ :SENSe]:CORRection:CSET<n>:DATA  
<freq>,<rel_ampl>{,<freq>,<rel_ampl>}
```

In the command above, you can ignore parameters {,<freq>,<rel\_ampl>} or assign them one or more values.

### 2. Vertical Bar |

The vertical bar separates two or more optional parameters. When you send a command, at least one of the parameters should be selected, such as

```
:DISPlay:MENU:STATe OFF|ON|0|1
```

In the command above, you can select "OFF", "ON", "0" or "1".

### 3. Square Brackets [ ]

The contents such as keywords enclosed in square brackets could be omitted, but all of them would be executed regardless whether they were omitted such as:

```
[ :SENSe]:CORRection:OFFSet[:MAGNitude]?
```

The following three commands serve a similar purpose:

```
:CORRection:OFFSet?
```

```
:CORRection:OFFSet:MAGNitude?
```

```
:SENSe:CORRection:OFFSet?
```

### 4. Triangle Brackets < >

The parameter enclosed in triangle brackets must be an effective value such as:

```
:DISPlay:BRIGtness <integer>
```

```
:DISPlay:BRIGtness 10
```

## Parameter Type

The commands presented in this manual contains 6 kinds of parameters and different parameters has different setting methods.

### 1. Boolean

The parameter should be "OFF", "ON", "0" or "1", such as  
:DISPlay:MENU:STATe OFF|ON|0|1

### 2. Keywords

The parameter should be an option listed in the command, such as  
:DISPlay:AFUnction:POSition BOTTOm|CENTer|TOP  
The parameter is "BOTTOm", "CENTer" or "TOP".

### 3. Integer

The parameter can be any integer within valid range unless where noted. Please do not use decimal for the parameter, or errors may occur, such as  
:DISPlay:BRIGtness <integer>  
Parameter <integer> can be any integer within 0 and 255.

### 4. Consecutive Real Number

The parameters can be any value within valid range (the default accuracy contains up to 6 digits after the decimal points), such as  
:CALCulate:BANDwidth:NDB <rel\_ampl>  
Parameter <rel\_ampl> can be any real number within -100 and 100.

### 5. Discrete

The parameters are discontinuous and should be a specified value, such as  
:CALCulate:MARKer<n>:MAXimum:MAX  
Parameter <n> can only be 1, 2, 3 or 4.

### 6. ASCII Character String

The parameter should be a composition of ASCII characters, such as  
:SYSTem:DATE <year>,<month>,<day>  
Parameters above should be a date string.

## Command Abbreviation

Since all commands are case-insensitive, you can use any kind of them. But if use an abbreviation, the capital letters specified in the command must be written completely.

For example:

:CALCulate:BANDwidth:NDB?

also could be:

:CALC:BAND:NDB?



## Chapter 2 Command Systems

This chapter gives you a detailed introduction to the sub-commands of DSA1000 series by A to Z. Two models are provided by the DSA1000 series: DSA1030 and DSA1020. Those commands related to the functions of preamplifier, demodulation, advanced measurement and tracking generator, only apply to the DSA1030 with relevant option installed. For more detail, refer to the remarks in each command subsystems.

- IEEE 488.2
- :ABORt
- :CALCulate
- :CALibration
- :CONFigure
- :COUPle
- :DISPlay
- :FETCh
- :FORMat
- :HCOPy
- :INITiate
- :INPut
- :MMEMory
- :OUTPut
- :READ
- [:SENSe]
- :SOURce
- :STATus
- :SYSTem
- :TRACe
- :TRIGer
- :UNIT

NOTE: In this command system, a query will return "N/A" (without double quotation marks) if the function to be queried is an uninstalled option or "ERR" (without double quotation marks) if the function is OFF or does not match unless where noted in this manual.

## IEEE 488.2

IEEE standard defines some common commands for users to check basic information about the instrument and execute common operations. These commands begin with a "\*" and have a keywords 3 characters length.

- \*CLS
- \*ESE
- \*ESE?
- \*ESR?
- \*IDN?
- \*OPC
- \*OPC?
- \*RST
- \*SRE
- \*SRE?
- \*STB?
- \*TRG
- \*WAI

<b>1. *CLS</b>	
Syntax	*CLS
Function	Clears all event registers and the error queue as well as the input queue.
<b>2. *ESE</b>	
Syntax	*ESE <value> *ESE?
Function	Sets the enable value of a standard event, such as *ESE 0 The query returns "0".
Explanations	Bit 1 and bit 6 are not used in the Standard Event Register and always regarded as "0", therefore <value> is a decimal number that relates to any binary number whose bits of 6 and 1 are both 0 within 00000000 (0) and 11111111 (255).
*Remark: Double quotation marks would't be returned after a query unless where noted in this manual.	
<b>3. *ESR?</b>	
Syntax	*ESR?
Function	Queries the standard event status register. The query returns a decimal number that relates to any binary number whose bits of 6 and 1 are both 0 within 00000000 (0) and 11111111 (255).
<b>4. *IDN?</b>	
Syntax	*IDN?
Function	Queries the instrument ID and returns a string with the length at least 35 characters.
<b>5. *OPC</b>	
Syntax	*OPC *OPC?
Function	*OPC: sets the "Operation Complete" (bit 0) of the standard event status register to "1" after the current operation. *OPC?: queries if the current operation has been finished or not, and returns "1" if successful; otherwise returns "0".

<b>6. *RST</b>	
Syntax	*RST
Function	Resets the analyzer to Default.
<b>7. *SRE</b>	
Syntax	*SRE <value> *SRE?
Function	Sets the service request enable value, such as *SRE 0 The query returns "0".
Explanations	Bit 0 and bit 1 are not used in the Status Byte Register and always regarded as "0", thereinto <value> a decimal number that relates to any binary number whose bits of 0 and 1 are both 0 within 00000000 (0) and 11111111 (255).
<b>8. *STB?</b>	
Syntax	*STB?
Function	Read Status Byte Query. The query returns a decimal number that relates to any binary number whose bits of 0 and 1 are both 0 within 00000000 (0) and 11111111 (255).
<b>9. *TRG</b>	
Syntax	*TRG
Function	Triggers a sweep or measurement immediately.
<b>10. *WAI</b>	
Syntax	*WAI
Function	Waits until done.

## :ABORT

- :ABORT

1. :ABORT	
Syntax	:ABORT
Function	Aborts the current operation and sweeps all over again.

## :CALCulate

- :CALCulate:BANDwidth:NDB
- :CALCulate:BANDwidth:RESult?
- :CALCulate:LLINe:ALL:DELeTe\*
- :CALCulate:LLINe:CONTRol:DOMain\*
- :CALCulate:LLINe<n>:CONTRol:INTErpolate:TYPE\*
- :CALCulate:LLINe<n>:STATe\*
- :CALCulate:LLINe<n>:DATA\*
- :CALCulate:LLINe<n>:DATA:MERGe\*
- :CALCulate:LLINe<n>:DELeTe\*
- :CALCulate:LLINe:FAIL?\*
- :CALCulate:LLINe:FAIL:STOP:STATe\*
- :CALCulate:LLINe:FAIL:RATIO?\*
- :CALCulate:MARKer:AOFF
- :CALCulate:MARKer<n>:CPEak[:STATe]
- :CALCulate:MARKer<n>:DELTA[:SET]:CENTer
- :CALCulate:MARKer<n>:DELTA[:SET]:SPAN
- :CALCulate:MARKer:FCOunt:RESolution
- :CALCulate:MARKer:FCOunt:RESolution:AUTO
- :CALCulate:MARKer:FCOunt[:STATe]
- :CALCulate:MARKer:FCOunt:X?
- :CALCulate:MARKer<n>:FUNCTion
- :CALCulate:MARKer<n>:MAXimum:MAX
- :CALCulate:MARKer<n>:MAXimum:LEFT
- :CALCulate:MARKer<n>:MAXimum:NEXT
- :CALCulate:MARKer<n>:MAXimum:RIGHT
- :CALCulate:MARKer<n>:MINimum
- :CALCulate:MARKer<n>:MODE
- :CALCulate:MARKer<n>:PEAK:EXCURsion
- :CALCulate:MARKer<n>:PEAK:SEARCh:MODE
- :CALCulate:MARKer<n>:PEAK[:SET]:CF
- :CALCulate:MARKer<n>:PEAK:THREshold
- :CALCulate:MARKer<n>:PTPeak
- :CALCulate:MARKer<n>[:SET]:CENTer
- :CALCulate:MARKer<n>[:SET]:RLEVel
- :CALCulate:MARKer<n>[:SET]:START

- :CALCulate:MARKer<n>[:SET]:STEP
- :CALCulate:MARKer<n>[:SET]:STOP
- :CALCulate:MARKer<n>:STATe
- :CALCulate:MARKer:TABLE:STATe
- :CALCulate:MARKer<n>:TRACe
- :CALCulate:MARKer<n>:TRACe:AUTO
- :CALCulate:MARKer:TRCKing[:STATe]
- :CALCulate:MARKer<n>:X
- :CALCulate:MARKer<n>:X:CENTer
- :CALCulate:MARKer<n>:X:POSition
- :CALCulate:MARKer<n>:X:POSition:CENTer
- :CALCulate:MARKer<n>:X:POSition:SPAN
- :CALCulate:MARKer<n>:X:POSition:START
- :CALCulate:MARKer<n>:X:POSition:STOP
- :CALCulate:MARKer<n>:X:READout
- :CALCulate:MARKer<n>:X:SPAN
- :CALCulate:MARKer<n>:X:START
- :CALCulate:MARKer<n>:X:STOP
- :CALCulate:MARKer<n>:Y?
- :CALCulate:NTData[:STATe]

**NOTE: The commands marked with "\*" only apply to DSA1030 which has been installed the advanced measurement option.**

<b>1. :CALCulate:BANDwidth:NDB</b>	
Syntax	:CALCulate:BANDwidth:NDB <rel_ampl> :CALCulate:BANDwidth:NDB?
Function	Sets the N value of the N dB BW measurement, such as :CALC:BAND:NDB -4 The query returns "-4.000000E+00".
Explanations	<rel_ampl> ranges from -100 dB to 100 dB.
Default	-3 dB
Front Panel	Marker Fctn, N dB BW
<b>2. :CALCulate:BANDwidth:RESult?</b>	
Syntax	:CALCulate:BANDwidth:RESult?
Function	The query returns the measured bandwidth (in Hz) according to the N value specified by command :CALCulate:BANDwidth:NDB, such as 80000000.
Explanations	The query returns "---" if the returned value is unavailable.
Front Panel	Marker Fctn, N dB BW
<b>3. :CALCulate:LLINE:ALL:DELeTe</b>	
Syntax	:CALCulate:LLINE:ALL:DELeTe
Function	Deletes the current edited limit line.
Front Panel	Meas, Pass/Fail Meas Setup, Limit, Upper, Del Limit Meas Setup, Limit, Lower, Del Limit
<b>4. :CALCulate:LLINE:CONTRol:DOMain</b>	
Syntax	:CALCulate:LLINE:CONTRol:DOMain FREQUency TIME :CALCulate:LLINE:CONTRol:DOMain?
Function	Specifies the unit of Frequency or Time as the unit of X-axis. The query returns "FREQ" or "TIME".
Explanations	<ul style="list-style-type: none"> <li>● The command works for both upper and lower limit.</li> <li>● Changes in unit of X-axis may delete all edited points under the current limit line.</li> </ul>
Default	FREQUency
Front Panel	Meas, Pass/Fail Meas Setup, X-axis, Freq/Time

<b>5. :CALCulate:LLINE&lt;n&gt;:CONTrol:INTerpolate:TYPE</b>	
Syntax	:CALCulate:LLINE<n>:CONTrol:INTerpolate:TYPE LOGarithmic LINEar :CALCulate:LLINE<n>:CONTrol:INTerpolate:TYPE?
Function	Sets the type of frequency interpolation. The query returns "LOG" or "LIN".
Explanations	<ul style="list-style-type: none"> <li>• &lt;n&gt; is 1 (lower limit line) or 2 (upper limit line).</li> <li>• In Log mode, both frequency and amplitude use Log for an interpolation operation.</li> <li>• In Line mode, the frequency uses Lin and the amplitude uses Log for an interpolation operation.</li> </ul>
Default	LINEar
Front Panel	Meas, Pass/Fail Meas Setup, Freq Interp, Log/Line
<b>6. :CALCulate:LLINE&lt;n&gt;:STATe</b>	
Syntax	:CALCulate:LLINE<n>:STATe OFF ON 0 1 :CALCulate:LLINE<n>:STATe?
Function	Turns on or off the upper/lower limit line. The query returns "0" or "1".
Explanations	<n> is 1 (lower limit line) or 2 (upper limit line).
Default	OFF 0
Front Panel	Meas, Pass/Fail Meas Setup, Limit, Upper/Lower, Test, On/Off
<b>7. :CALCulate:LLINE&lt;n&gt;:DATA</b>	
Syntax	:CALCulate:LLINE<n>:DATA <x-axis>,<ampl>,<connected>{,<x-axis>,<ampl>,<connected>} :CALCulate:LLINE<n>:DATA?
Function	Creates a limit line, take the case of three points: :CALC:LLIN2:DATA 50,100,0,100,150,1,200,200,1 The query returns: 50,100.000000,0,100,150.000000,1,200,200.000000,1
Explanations	<ul style="list-style-type: none"> <li>• &lt;n&gt;: denotes the limit line to be edited, which can be "1" (lower limit line) or "2" (upper limit line).</li> <li>• &lt;x-axis&gt;: denotes the frequency (in Hz) or time (in us) of selected point.</li> </ul>

	<ul style="list-style-type: none"> <li>● &lt;ampl&gt;: denotes the amplitude (in dBm) of selected point.</li> <li>● &lt;connected&gt;: denotes if to connect the points of current with the last one, which can be 1(connect) or 0 (disconnect).</li> <li>● Each limit line allows no more than 200 points.</li> <li>● The query returns "NULL" if none of points are created for the specified limit line.</li> </ul>
Front Panel	Meas, Pass/Fail Meas Setup, Edit
<b>8. :CALCulate:LLINE&lt;n&gt;:DATA:MERGE</b>	
Syntax	:CALCulate:LLINE<n>:DATA:MERGe <x-axis>,<ampl>,<connected>{,<x-axis>,<ampl>,<connected>}
Function	Adds extra points to the limit line, such as :CALC:LLIN2:DATA:MERG 250,200,1,300,250,1
Explanations	Refer to :CALCulate:LLINE<n>:DATA to get explanation of the parameters in this command.
<b>9. :CALCulate:LLINE&lt;n&gt;:DELeTe</b>	
Syntax	:CALCulate:LLINE<n>:DELeTe
Function	Deletes the limit line you are editing.
Explanations	<n> is 1 (lower limit) or 2 (upper limit).
Front Panel	Meas, Pass/Fail Meas Setup, Limit, Upper, Del Limit Meas Setup, Limit, Lower, Del Limit
<b>10. :CALCulate:LLINE:FAIL?</b>	
Syntax	:CALCulate:LLINE:FAIL?
Function	Queries the Pass/Fail test result. The query returns "PASS" or "FAIL", or "UNMEAS" if the measurement is unfinished.
<b>11. :CALCulate:LLINE:FAIL:STOP:STATE</b>	
Syntax	:CALCulate:LLINE:FAIL:STOP:STATe OFF ON 0 1 :CALCulate:LLINE:FAIL:STOP:STATe?
Function	Sets whether to continue a new sweep or not. The query returns "0" or "1".
Default	ON 1
Front Panel	Meas, Pass/Fail

	Meas Setup, Fail Stop, On/Off
<b>12. :CALCulate:LLINe:FAIL:RATIo?</b>	
Syntax	:CALCulate:LLINe:FAIL:RATIo?
Function	The query returns the failure rate from the test results, such as 0.000000E+00.
<b>13. :CALCulate:MARKer:AOff</b>	
Syntax	:CALCulate:MARKer:AOff
Function	Turns off all markers and functions based upon markers.
Front Panel	Marker, All Off
<b>14. :CALCulate:MARKer&lt;n&gt;:CPEak[:STATe]</b>	
Syntax	:CALCulate:MARKer<n>:CPEak[:STATe] OFF ON 0 1 :CALCulate:MARKer<n>:CPEak[:STATe]?
Function	Turns on or off the Cont Peak Search. The query returns "0" or "1".
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● The command turns on Marker 1 automatically if none is enabled.</li> </ul>
Default	OFF 0
Front Panel	Peak, Cont Peak, On/Off
<b>15. :CALCulate:MARKer&lt;n&gt;:DELTA[:SET]:CENTer</b>	
Syntax	:CALCulate:MARKer<n>:DELTA[:SET]:CENTer
Function	Sets the center frequency to the frequency difference between the two markers in "Delta", "Delta Pair" or "Span Pair" marker.
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● Sending :CALCulate:MARKer&lt;n&gt;:MODE selects a marker type from "Delta", "Delta Pair" or "Span Pair".</li> <li>● The command is invalid in Zero-span mode.</li> </ul>
Front Panel	Marker, Delta/Delta Pair/Span Pair Marker ->, MkrΔ->CF
<b>16. :CALCulate:MARKer&lt;n&gt;:DELTA[:SET]:SPAN</b>	
Syntax	:CALCulate:MARKer<n>:DELTA[:SET]:SPAN
Function	Sets the analyzer span to the frequency difference between the two

	markers in "Delta", "Delta Pair" and "Span Pair" marker.
Explanations	<ul style="list-style-type: none"> <li>• &lt;n&gt; is 1, 2, 3 or 4.</li> <li>• Sending :CALCulate:MARKer&lt;n&gt;:MODE selects a marker type from "Delta", "Delta Pair" or "Span Pair".</li> <li>• The command is invalid in Zero-span mode.</li> </ul>
Front Panel	Marker, Delta/Delta Pair/Span Pair Marker ->, MkrΔ ->Span
<b>17. :CALCulate:MARKer:FCOunt:RESolution</b>	
Syntax	:CALCulate:MARKer:FCOunt:RESolution <freq> :CALCulate:MARKer:FCOunt:RESolution?
Function	Sets the frequency counter resolution, the default unit is Hz, such as :CALC:MARK:FCO:RES 1000 or :CALC:MARK:FCO:RES 1KHZ The query returns "1000".
Explanations	<ul style="list-style-type: none"> <li>• &lt;freq&gt; ranges from 1 Hz to 100 kHz (1Hz, 10Hz, 100Hz, 1kHz, 10kHz, 100kHz)</li> <li>• Sending :CALCulate:MARKer:FCOunt:RESolution:AUTO can set the resolution mode to Auto.</li> </ul>
Default	1000 (1kHz)
Front Panel	Marker Fctn, Freq Count, Resolution, Manual
<b>18. :CALCulate:MARKer:FCOunt:RESolution:AUTO</b>	
Syntax	:CALCulate:MARKer:FCOunt:RESolution:AUTO OFF ON 0 1 :CALCulate:MARKer:FCOunt:RESolution:AUTO?
Function	Set the resolution mode of frequency counter to Auto in order to get a faster and accurately counting in connection with specified span. The query returns "0" or "1".
Default	ON 1
Front Panel	Marker Fctn, Freq Count, Resolution, Auto
<b>19. :CALCulate:MARKer:FCOunt[:STATE]</b>	
Syntax	:CALCulate:MARKer:FCOunt[:STATE] OFF ON 0 1 :CALCulate:MARKer:FCOunt[:STATE]?
Function	Turns on or off the frequency counter. The query returns "0" or "1".
Explanations	<ul style="list-style-type: none"> <li>• Sending :CALCulate:MARKer:FCOunt:X? queries the readout of the frequency counter.</li> </ul>

	<ul style="list-style-type: none"> <li>The command will refer all activated markers on condition that the frequency counter is enabled.</li> </ul>
Default	OFF 0
Front Panel	Marker Fctn, Freq Count, State, On/Off
<b>20. :CALCulate:MARKer:FCOunt:X?</b>	
Syntax	:CALCulate:MARKer:FCOunt:X?
Function	The query returns the readout of the frequency counter, such as 1500335500, (in Hz).
Explanations	The query returns "9000000000000000 (9e15)" when the frequency counter is disabled.
<b>21. :CALCulate:MARKer&lt;n&gt;:FUNctIon</b>	
Syntax	:CALCulate:MARKer<n>:FUNctIon NDB NOISe OFF :CALCulate:MARKer<n>:FUNctIon?
Function	Selects a measurement type for the specified marker. The query returns "NDB", "NOIS" or "OFF".
Explanations	<ul style="list-style-type: none"> <li>&lt;n&gt; is 1, 2, 3 or 4.</li> <li>NDB: N dB BW measurement.</li> <li>NOISe: Noise measurement.</li> <li>OFF: turns off all measurements.</li> </ul>
Default	OFF 0
Front Panel	Marker Fctn, Noise Mkr/N dB BW/ Function Off
<b>22. :CALCulate:MARKer&lt;n&gt;:MAXimum:MAX</b>	
Syntax	:CALCulate:MARKer<n>:MAXimum:MAX
Function	Executes a peak search according to the mode selected by :CALCulate:MARKer<n>:PEAK:SEARch:MODE.
Explanations	<n> is 1, 2, 3 or 4.
Front Panel	Peak, Search Para, Peak Search, Max
<b>23. :CALCulate:MARKer&lt;n&gt;:MAXimum:LEFT</b>	
Syntax	:CALCulate:MARKer<n>:MAXimum:LEFT
Function	Searches the nearest peak located to the left side of the current peak and meets the searching condition. The peak is then identified with a marker.
Explanations	<ul style="list-style-type: none"> <li>&lt;n&gt; is 1, 2, 3 or 4.</li> </ul>

	<ul style="list-style-type: none"> <li>● The peak which has a smaller frequency compared with 1% of span or the resolution bandwidth will be ignored.</li> <li>● When no desired peak is found, the system will show "No peak found".</li> </ul>
Front Panel	Method one: Peak, Peak Left Method two: L-Peak
<b>24. :CALCulate:MARKer&lt;n&gt;:MAXimum:NEXT</b>	
Syntax	:CALCulate:MARKer<n>:MAXimum:NEXT
Function	Searches the peak whose amplitude is the closest to the current peak's and meets the searching condition. The peak is then identified with a marker
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● The peak which has a smaller frequency compared with 1% of span or the resolution bandwidth will be ignored.</li> <li>● When no desired peak is found, the system will show "No peak found".</li> </ul>
Front Panel	Peak, Next Peak
<b>25. :CALCulate:MARKer&lt;n&gt;:MAXimum:RIGHT</b>	
Syntax	:CALCulate:MARKer<n>:MAXimum:RIGHT
Function	Searches the nearest peak located to the right side of the current peak and meets the searching condition. The peak is then identified with a marker.
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● The peak which has a smaller frequency compared with 1% of span or the resolution bandwidth will be ignored.</li> <li>● When no desired peak is found, the system will show "No peak found".</li> </ul>
Front Panel	Method one: Peak, Peak Right Method two: R-Peak
<b>26. :CALCulate:MARKer&lt;n&gt;:MINimum</b>	
Syntax	:CALCulate:MARKer<n>:MINimum
Function	Searches the peak with the minimum amplitude on the trace and identifies it with a marker.
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> </ul>

	<ul style="list-style-type: none"> <li>● The peak which has a smaller frequency compared with 1% of span or the resolution bandwidth will be ignored.</li> <li>● When no desired peak is found, the system will show "No peak found".</li> </ul>
Front Panel	Peak, Min Search
<b>27. :CALCulate:MARKer&lt;n&gt;:MODE</b>	
Syntax	:CALCulate:MARKer<n>:MODE POSition DELTA BAND SPAN :CALCulate:MARKer<n>:MODE?
Function	Sets the mode of the selected marker. Queries the specified marker mode and returns "POS", "DELTA", "BAND" or "SPAN".
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● POSition: Normal</li> <li>● DELTA: Delta</li> <li>● BAND: Delta Pair</li> <li>● SPAN: Span Pair</li> </ul>
Default	POSition
Front Panel	Marker, Normal/Delta/Delta Pair/Span Pair
<b>28. :CALCulate:MARKer&lt;n&gt;:PEAK:EXCursion</b>	
Syntax	:CALCulate:MARKer<n>:PEAK:EXCursion <rel_ampl> :CALCulate:MARKer<n>:PEAK:EXCursion?
Function	Sets the excursion between the specified peak and the minimum amplitude on both sides of it in dB, such as :CALC:MARK1:PEAK:EXC 12 The query returns "1.200000E+01".
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● &lt;rel_ampl&gt; ranges from 0 to 200 dB.</li> </ul>
Default	10 dB
Front Panel	Peak, Search Para, Pk Excursn
<b>29. :CALCulate:MARKer&lt;n&gt;:PEAK:SEARch:MODE</b>	
Syntax	:CALCulate:MARKer<n>:PEAK:SEARch:MODE PARAmeter MAXimum :CALCulate:MARKer<n>:PEAK:SEARch:MODE?
Function	Sets the mode of Peak Search.

	The query returns "PAR" or "MAX".
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● PARAmeter: searches the peak that meets the condition of Search Para and identifies it with a marker.</li> <li>● MAXimum: searches the miximum on the trace and identifies it with a marker.</li> <li>● An error message may generate if no desired peak is found.</li> <li>● The command has meaning only for the peak search which will be executed when pressing the <b>Peak</b> key on the front panel regardless of Next Peak, Peak Right, Peak Left and Min Search.</li> </ul>
Default	MAXimum
Front Panel	Peak, Search Para, Peak Search, Max/Param
<b>30. :CALCulate:MARKer&lt;n&gt;:PEAK[:SET]:CF</b>	
Syntax	:CALCulate:MARKer<n>:PEAK[:SET]:CF
Function	Executes a peak search and sets the frequency of the current peak to the center frequency of the analyzer.
Explanations	<n> is 1, 2, 3 or 4.
Front Panel	FREQ, Peak->CF
<b>31. :CALCulate:MARKer&lt;n&gt;:PEAK:THReshold</b>	
Syntax	:CALCulate:MARKer<n>:PEAK:THReshold <ampl> :CALCulate:MARKer<n>:PEAK:THReshold?
Function	Sets the peak limit, the default unit is dBm, such as :CALC:MARK1:PEAK:THR -100 The query returns "-1.000000E+02".
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● &lt;ampl&gt; ranges from -200 dBm to 0 dBm.</li> <li>● A real peak should be greater than the peak limit.</li> </ul>
Default	-90 dBm
Front Panel	Peak, Search Para, Pk Thresh
<b>32. :CALCulate:MARKer&lt;n&gt;:PTPeak</b>	
Syntax	:CALCulate:MARKer<n>:PTPeak
Function	Executes the peak search.
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● Sending this command automatically changes the marker from</li> </ul>

	"Normal" to "Delta Pair".
Front Panel	Peak, Peak Search
<b>33. :CALCulate:MARKer&lt;n&gt;[:SET]:CENTer</b>	
Syntax	:CALCulate:MARKer<n>[:SET]:CENTer
Function	Sets the center frequency based upon the current marker frequency.
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● In "Normal" marker mode, the center frequency is the frequency at which the marker is located.</li> <li>● In "Delta" marker mode, the center frequency is the frequency at which the delta marker is located.</li> </ul>
Front Panel	Marker →, MkrΔ->CF
<b>34. :CALCulate:MARKer&lt;n&gt;[:SET]:RLEVel</b>	
Syntax	:CALCulate:MARKer<n>[:SET]:RLEVel
Function	Sets the reference level of the analyzer according to the current marker amplitude.
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● In "Normal" marker mode, the reference level is the amplitude at which the marker is located.</li> <li>● In "Delta" marker mode, the reference level is the amplitude at which the delta marker is located.</li> </ul>
Front Panel	Marker →, Mkr->Ref
<b>35. :CALCulate:MARKer&lt;n&gt;[:SET]:START</b>	
Syntax	:CALCulate:MARKer<n>[:SET]:START
Function	Sets the start frequency of the analyzer according to the current marker frequency.
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● In "Normal" marker mode, the start frequency is the frequency at which the marker is located.</li> <li>● In "Delta" marker mode, the start frequency is the frequency at which the delta marker is located.</li> <li>● The command is invalid in Zero-span mode.</li> </ul>
Front Panel	Marker →, Mkr->Start
<b>36. :CALCulate:MARKer&lt;n&gt;[:SET]:STEP</b>	

Syntax	:CALCulate:MARKer<n>[:SET]:STEP
Function	Sets the center frequency step of the analyzer according to the current marker frequency.
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● In "Normal" marker mode, the center frequency is the frequency at which the marker is located.</li> <li>● In "Delta" marker mode, the center frequency step is the frequency at which the delta marker is located.</li> <li>● The command can also be used to get the information about the next harmonic when in harmonic measurement.</li> <li>● The command is invalid in Zero-span mode.</li> </ul>
Front Panel	Marker →, Mkr-→Step

**37. :CALCulate:MARKer<n>[:SET]:STOP**

Syntax	:CALCulate:MARKer<n>[:SET]:STOP
Function	Sets the stop frequency of the analyzer according to the current marker frequency.
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● In "Normal" marker mode, the stop frequency is the frequency at which the marker is located.</li> <li>● In "Delta" marker mode, the stop frequency is the frequency at which the delta marker is located.</li> <li>● The command is invalid in Zero-span mode.</li> </ul>
Front Panel	Marker →, Mkr-→Stop

**38. :CALCulate:MARKer<n>:STATe**

Syntax	:CALCulate:MARKer<n>:STATe OFF ON 0 1 :CALCulate:MARKer<n>:STATe?
Function	Turns on or off the selected marker. The query returns "0" or "1".
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● If there is no opened marker, turning on a marker uses the "POSITION" (Normal mode) as the default.</li> </ul>
Default	OFF 0
Front Panel	Marker, Select Mkr 1/2/3/4 Marker, Off

<b>39. :CALCulate:MARKer:TABLE:STATE</b>	
Syntax	:CALCulate:MARKer:TABLE:STATE OFF ON 0 1 :CALCulate:MARKer:TABLE:STATE?
Function	Turns on or off the marker table. The query returns "0" or "1".
Default	OFF 0
Front Panel	Marker, Mkr Table, On/Off
<b>40. :CALCulate:MARKer&lt;n&gt;:TRACe</b>	
Syntax	:CALCulate:MARKer<n>:TRACe <integer> :CALCulate:MARKer<n>:TRACe?
Function	Assigns a trace for the specified marker from Trace 1, Trace 2, Trace 3 or Trace 4 (math trace), such as :CALC:MARK1:TRAC 1 The query returns "1".
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● &lt;integer&gt; is 1, 2, 3 or 4 (math trace). The trace specified here must be On.</li> <li>● The query returns the trace number if the trace mode is Auto.</li> </ul>
Default	1
Front Panel	Marker, Mkr Trace, 1/2/3/Math
<b>41. :CALCulate:MARKer&lt;n&gt;:TRACe:AUTO</b>	
Syntax	:CALCulate:MARKer<n>:TRACe:AUTO OFF ON 0 1 :CALCulate:MARKer<n>:TRACe:AUTO?
Function	Turns on or off the auto trace function for the specified marker. The query returns "0" or "1".
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● The current marker stands on the corresponding trace if Auto is disabled.</li> </ul>
Default	ON 1
Front Panel	Marker, Mkr Trace, Auto
<b>42. :CALCulate:MARKer:TRCKing[:STATE]</b>	
Syntax	:CALCulate:MARKer:TRCKing[:STATE] OFF ON 0 1 :CALCulate:MARKer:TRCKing[:STATE]?
Function	Turns on or off the signal track function.

	The query returns "0" or "1".
Explanations	When the signal track function is On, the analyzer executes a peak search, sets the current peak frequency to the center frequency after each sweep and holds the signal at the center of screen all the time.
Default	OFF 0
Front Panel	FREQ, Signal Track, State, On/Off
<b>43. :CALCulate:MARKer&lt;n&gt;:X</b>	
Syntax	:CALCulate:MARKer<n>:X <param> :CALCulate:MARKer<n>:X?
Function	Sets the X-axis value at which the marker is located, the default unit is Hz, such as :CALC:MARK1:X 15000000 or :CALC:MARK1:X 1.5GHZ The query returns "15000000".
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● &lt;param&gt; is within the range of "X": If current readout mode is "Frequency", the range cannot exceed the sweep frequency and unit input of Hz (default), kHz, MHz or GHz is available. If current readout mode is "Δtime", the range cannot exceed the sweep time and unit input of s (default), us, ms or ks is available.</li> <li>● In "Normal" marker mode, the command sets the X value at which the current marker is located.</li> <li>● In "Delta" marker mode, the command sets the X value of the current delta marker according to the reference marker.</li> <li>● In "Delta Pair" marker mode, the command sets the X value of the reference marker or the delta marker according to the current menu status.</li> <li>● In "Span Pair" marker mode, the command sets the X value of the span between or center of the reference marker and delta marker according to the current menu status.</li> </ul>
Default Unit	Corresponds with the trace unit of the current marker.
Front Panel	Marker, Normal Marker, Delta Marker, Delta Pair, Ref/Delta

	Marker, Span Pair, Span/Center
<b>44. :CALCulate:MARKer&lt;n&gt;:X:CENTer</b>	
Syntax	:CALCulate:MARKer<n>:X:CENTer <param> :CALCulate:MARKer<n>:X:CENTer?
Function	Sets the X value at the center of the reference marker and the delta marker in "Span Pair" marker mode, the default unit is Hz, such as :CALC:MARK1:X:CENT 405000000 or :CALC:MARK1:X:CENT 405MHZ The query returns "405000000".
Explanations	<ul style="list-style-type: none"> <li>• &lt;n&gt; is 1, 2, 3 or 4.</li> <li>• &lt;param&gt; is within the range of X.</li> </ul>
Front Panel	Marker, Span Pair, Span/Center
<b>45. :CALCulate:MARKer&lt;n&gt;:X:POSition</b>	
Syntax	:CALCulate:MARKer<n>:X:POSition <integer> :CALCulate:MARKer<n>:X:POSition?
Function	Sets the position of the "Normal" marker on the trace, such as :CALC:MARK1:X:POSition 100 The query returns "100".
Explanations	<ul style="list-style-type: none"> <li>• &lt;n&gt; is 1, 2, 3 or 4.</li> <li>• For the range of &lt;integer&gt;, refer to "[[:SENSe]:SWEep:POINts]".</li> </ul>
<b>46. :CALCulate:MARKer&lt;n&gt;:X:POSition:CENTer</b>	
Syntax	:CALCulate:MARKer<n>:X:POSition:CENTer <param> :CALCulate:MARKer<n>:X:POSition:CENTer?
Function	Sets the position of the center frequency of "Span Pair" marker on the trace, such as :CALC:MARK1:X:POS:CENTer 200 The query returns "200".
Explanations	<ul style="list-style-type: none"> <li>• &lt;n&gt; is 1, 2, 3 or 4.</li> <li>• For the range of &lt;param&gt;, refer to "[[:SENSe]:SWEep:POINts]".</li> </ul>
<b>47. :CALCulate:MARKer&lt;n&gt;:X:POSition:SPAN</b>	
Syntax	:CALCulate:MARKer<n>:X:POSition:SPAN <param> :CALCulate:MARKer<n>:X:POSition:SPAN?
Function	Sets the number of points on the trace. The trace is limited by the

	reference marker and the delta marker in "Span Pair" marker mode. Such as :CALC:MARK1:X:POS:SPAN 150 The query returns "150".
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● For the range of &lt;param&gt;, refer to "[ :SENSe]:SWEep:POINts".</li> </ul>
<b>48. :CALCulate:MARKer&lt;n&gt;:X:POsition:START</b>	
Syntax	:CALCulate:MARKer<n>:X:POsition:START <param> :CALCulate:MARKer<n>:X:POsition:START?
Function	Sets the position of the reference marker on the related trace in "Delta Pair" marker mode, such as :CALC:MARK1:X:POS:START 10 The query returns "10".
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● For the range of &lt;param&gt;, refer to "[ :SENSe]:SWEep:POINts".</li> </ul>
<b>49. :CALCulate:MARKer&lt;n&gt;:X:POsition:STOP</b>	
Syntax	:CALCulate:MARKer<n>:X:POsition:STOP <param> :CALCulate:MARKer<n>:X:POsition:STOP?
Function	Sets the position of the delta marker on the related trace in "Delta Pair" marker mode, such as :CALC:MARK1:X:POS:STOP 600 The query returns "600".
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● For the range of &lt;param&gt;, refer to "[ :SENSe]:SWEep:POINts".</li> </ul>
<b>50. :CALCulate:MARKer&lt;n&gt;:X:READout</b>	
Syntax	:CALCulate:MARKer<n>:X:READout FREQuency TIME ITIME PERiod :CALCulate:MARKer<n>:X:READout?
Function	Sets the marker readout type on X-axis. The query returns "FREQ", "TIME", "ITIM" or "PER".
Explanations	<n> is 1, 2, 3 or 4.
Default	FREQuency
Front Panel	Marker, Readout, Frequency/Period/ $\Delta$ Time/ $1/\Delta$ time
<b>51. :CALCulate:MARKer&lt;n&gt;:X:SPAN</b>	

Syntax	:CALCulate:MARKer<n>:X:SPAN <param> :CALCulate:MARKer<n>:X:SPAN?
Function	Sets the X value of the marker span on the related trace in "Span Pair" marker mode. The unit of X value corresponds to the unit of the current marker trace. Such as :CALC:MARK1:X:SPAN 500000000 The query returns "500000000".
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● &lt;param&gt; is within the range of X. If current readout mode is "Frequency", the range cannot exceed the sweep frequency and unit input of Hz (default), kHz, MHz or GHz is available. If current readout mode is "Δtime", the range cannot exceed the sweep time and unit input of s (default), us, ms or ks is available.</li> </ul>
Front Panel	Marker, Span Pair, Span
<b>52. :CALCulate:MARKer&lt;n&gt;:X:START</b>	
Syntax	:CALCulate:MARKer<n>:X:START <param> :CALCulate:MARKer<n>:X:START?
Function	Sets the X value of the reference marker on related trace in "Delta Pair" marker mode. The unit of X value corresponds to the unit of the current marker trace. Such as :CALC:MARK1:X:START 1320000000 The query returns "1320000000".
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● &lt;param&gt; is within the range of X. If current readout mode is "Frequency", the range cannot exceed the sweep frequency and unit input of Hz (default), kHz, MHz or GHz is available. If current readout mode is "Δtime", the range cannot exceed the sweep time and unit input of s (default), us, ms or ks is available.</li> </ul>
Front Panel	Marker, Delta Pair, Ref
<b>53. :CALCulate:MARKer&lt;n&gt;:X:STOP</b>	
Syntax	:CALCulate:MARKer<n>:X:STOP <param>

	:CALCulate:MARKer<n>:X:STOP?
Function	Sets the X value of the delta marker on related trace in "Delta Pair". The unit of the X value corresponds to the unit of the current marker trace. Such as :CALC:MARK1:X:STOP 1820000000 The query returns "1820000000".
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is 1, 2, 3 or 4.</li> <li>● &lt;param&gt; is within the range of X. If current readout mode is "Frequency", the range cannot exceed the sweep frequency and unit input of Hz (default), kHz, MHz or GHz is available.</li> <li>● If current readout mode is "Δtime", the range cannot exceed the sweep time and unit input of s (default), us, ms or ks is available.</li> </ul>
Front Panel	Marker, Delta Pair, Delta
<b>54. :CALCulate:MARKer&lt;n&gt;:Y?</b>	
Syntax	:CALCulate:MARKer<n>:Y?
Function	The query returns the Y-axis readout of the specified marker, the default unit is dBm, such as -2.697830E-01.
<b>55. :CALCulate:NTData[:STATe]</b>	
Syntax	:CALCulate:NTData[:STATe] OFF ON 0 1 :CALCulate:NTData[:STATe]?
Function	Turns on or off the Normalization function. The query returns "0" or "1".
Default	OFF 0

## :CALibration

- :CALibration:[ALL]
- :CALibration:AUTO

<b>1. :CALibration:[ALL]</b>	
Syntax	:CALibration[:ALL]
Function	Calibrates immediately.
Front Panel	System, Calibrate, Cal Now
<b>2. :CALibration:AUTO</b>	
Syntax	:CALibration:AUTO OFF ON 0 1 :CALibration:AUTO?
Function	Enables or disables the Auto calibration. The query returns "0" or "1".
Default	ON 1 (NOTE: The Auto calibration is set to On as default and will always hold the setting before a power circle.)
Front Panel	System, Calibrate, Self-Cal, On/Off

## :CONFigure

- :CONFigure:ACPower
- :CONFigure:CHPower
- :CONFigure:CNRatio
- :CONFigure:EBWidth
- :CONFigure:HDISt
- :CONFigure:OBWidth
- :CONFigure:PF
- :CONFigure:SANalyzer
- :CONFigure:TOI
- :CONFigure:TPOWER
- :CONFigure?

**NOTE: The CONFigure commands only apply to DSA1030 which has been installed the advanced measurement option.**

<b>1. :CONFigure:ACPower</b>	
Syntax	:CONFigure:ACPower
Function	Enables the Adjacent Channel Power measurement.
Explanations	<ul style="list-style-type: none"> <li>● The command first restores the system parameters to factory defaults.</li> <li>● The command interrupts the current measurement and restores the analyzer to the defaults of specified measurement function.</li> <li>● You can use READ command to initialize a measurement function without any further settings of Default.</li> </ul>
Front Panel	Preset Meas, Meas Fctn, ACP
<b>2. :CONFigure:CHPower</b>	
Syntax	:CONFigure:CHPower
Function	Enables the Channel Power measurement.
Explanations	<ul style="list-style-type: none"> <li>● The command first restores the system parameters to factory defaults.</li> <li>● The command interrupts the current measurement and restores the analyzer to the defaults of specified measurement function.</li> <li>● You can use READ command to initialize a measurement function without any further settings of Default.</li> </ul>
Front Panel	Preset Meas, Meas Fctn, Chan Pwr
<b>3. :CONFigure:CNRatio</b>	
Syntax	:CONFigure:CNRatio
Function	Enables the C/N Ratio measurement.
Explanations	<ul style="list-style-type: none"> <li>● The command first restores the system parameters to factory defaults.</li> <li>● The command interrupts the current measurement and restores the analyzer to the defaults of specified measurement function.</li> <li>● You can use READ command to initialize a measurement function without any further settings of Default.</li> </ul>
Front Panel	Preset

	Meas, Meas Fctn, C/N Ratio
<b>4. :CONFigure:EBWidth</b>	
Syntax	:CONFigure:EBWidth
Function	Enables the Emission BandWidth measurement.
Explanations	<ul style="list-style-type: none"> <li>● The command first restores the system parameters to factory defaults.</li> <li>● The command interrupts the current measurement and restores the analyzer to the defaults of specified measurement function.</li> <li>● You can use READ command to initialize a measurement function without any further settings of Default.</li> </ul>
Front Panel	Preset Meas, Meas Fctn, EBW
<b>5. :CONFigure:HDISt</b>	
Syntax	:CONFigure:HDISt
Function	Enables the Harmonic Distortion measurement.
Explanations	<ul style="list-style-type: none"> <li>● The command first restores the system parameters to factory defaults.</li> <li>● The command interrupts the current measurement and restores the analyzer to the defaults of specified measurement function.</li> <li>● You can use READ command to initialize a measurement function without any further settings of Default.</li> </ul>
Front Panel	Preset Meas, Meas Fctn, Harmo Dist
<b>6. :CONFigure:OBWidth</b>	
Syntax	:CONFigure:OBWidth
Function	Enables the Occupied BandWidth measurement.
Explanations	<ul style="list-style-type: none"> <li>● The command first restores the system parameters to factory defaults.</li> <li>● The command interrupts the current measurement and restores the analyzer to the defaults of specified measurement function.</li> <li>● You can use READ command to initialize a measurement</li> </ul>

	function without any further settings of Default.
Front Panel	Preset Meas, Meas Fctn, OBW
<b>7. :CONFigure:PF</b>	
Syntax	:CONFigure:PF
Function	Enables the Pass/Fail measurement.
Explanations	<ul style="list-style-type: none"> <li>● The command first restores the system parameters to factory defaults.</li> <li>● The command interrupts the current measurement and restores the analyzer to the defaults of specified measurement function.</li> <li>● You can use READ command to initialize a measurement function without any further settings of Default.</li> </ul>
Front Panel	Preset Meas, Meas Fctn, Pass/Fail
<b>8. :CONFigure:SANalyzer</b>	
Syntax	:CONFigure:SANalyzer
Function	Disables the current measurement and enables the Sweep mode of the analyzer.
Explanations	<ul style="list-style-type: none"> <li>● The command first restores the system parameters to factory defaults.</li> <li>● The command interrupts the current measurement and restores the analyzer to the defaults of specified measurement function.</li> <li>● You can use READ command to initialize a measurement function without any further settings of Default.</li> </ul>
Front Panel	Preset Meas, Meas Fctn, Off
<b>9. :CONFigure:TOI</b>	
Syntax	:CONFigure:TOI
Function	Enables the Third Order Intermodulation measurement.
Explanations	<ul style="list-style-type: none"> <li>● The command first restores the system parameters to factory defaults.</li> <li>● The command interrupts the current measurement and</li> </ul>

	<p>restores the analyzer to the defaults of specified measurement function.</p> <ul style="list-style-type: none"> <li>You can use READ command to initialize a measurement function without any further settings of Default.</li> </ul>
Front Panel	<p>Preset Meas, Meas Fctn, TOI</p>
<b>10. :CONFigure:TPOWer</b>	
Syntax	:CONFigure:TPOWer
Function	Enables the Time-domain Power measurement.
Explanations	<ul style="list-style-type: none"> <li>The command first restores the system parameters to factory defaults.</li> <li>The command interrupts the current measurement and restores the analyzer to the defaults of specified measurement function.</li> <li>You can use READ command to initialize a measurement function without any further settings of Default.</li> </ul>
Front Panel	<p>Preset Meas, Meas Fctn, T-Power</p>
<b>11. :CONFigure?</b>	
Syntax	:CONFigure?
Function	Queries the current measurement.
Return Value	The query returns "OFF", "TPOW", "ACP", "CHP", "OBW", "EBW", "CNR", "HD", "TOI" or "PF".

## :COUPle

- :COUPle

<b>1. :COUPle</b>	
Syntax	:COUPle ALL NONE :COUPle?
Function	Sets all related parameters on the basis of their own coupling relation. The query returns "ALL" or "NONE".
Explanations	Following parameters are involved: <ul style="list-style-type: none"> <li>● CF Step</li> <li>● Attenuator</li> <li>● RBW (Resolution BandWidth)</li> <li>● VBW (Video BandWidth)</li> <li>● Sweep Time</li> </ul>
Default	ALL
Front Panel	System, Coupl Param

## :DISPlay

- :DISPlay:AFUnction:POSition
- :DISPlay:ANNotation:CLOCK
- :DISPlay:BRIGtness
- :DISPlay:ENABle
- :DISPlay:MENU:STATe
- :DISPlay:MENU:HTIME
- :DISPlay:MSGswitch:STATe
- :DISPlay:SKIN
- :DISPlay:SSAVer:TIME
- :DISPlay:WINdow:TRACe:GRATicule:GRID
- :DISPlay:WINdow:TRACe:Y:DLINe
- :DISPlay:WINdow:TRACe:Y:DLINe:STATe
- :DISPlay:WINdow:TRACe:Y:SCALE:PDIVision
- :DISPlay:WINdow:TRACe:Y:SCALE:RLEVel
- :DISPlay:WINdow:TRACe:Y:SCALE:RLEVel:OFFSet
- :DISPlay:WINdow:TRACe:Y:SCALE:SPACing
- :DISPlay:WINdow:TRACe:Y:SCALE:NRLevel\*\*\*
- :DISPlay:WINdow:TRACe:Y:SCALE:NRPosition\*\*\*

**NOTE: The commands marked with "\*\*\*" only apply to DSA1030 which has been installed the tracking generator option.**

<b>1. :DISPlay:AFUnction:POSition</b>	
Syntax	:DISPlay:AFUnction:POSition BOTTOm CENTer TOP :DISPlay:AFUnction:POSition?
Function	Sets the position of active function area. The query returns "BOTT", "CENT" or "TOP".
Explanations	This setting is persistent even through a power cycle.
Default	TOP
Front Panel	Display, Active Fctn, Top/Center/Bottom
<b>2. :DISPlay:ANNotation:CLOCK</b>	
Syntax	:DISPlay:ANNotation:CLOCK[:STATe] OFF ON 0 1 :DISPlay:ANNotation:CLOCK[:STATe]?
Function	Sets the clock status. The query returns "0" or "1".
Explanations	This setting is persistent even through a power cycle.
Default	ON 1
Front Panel	System, Time/Date, Time/Date, On/Off
<b>3. :DISPlay:BRIGtness</b>	
Syntax	:DISPlay:BRIGtness <integer> :DISPlay:BRIGtness?
Function	Sets the screen brightness, such as :DISP:BRIG 6 The query returns "6".
Explanations	<integer> ranges from 0 to 7.
Default	6
Front Panel	Display, Brightness
<b>4. :DISPlay:ENABle</b>	
Syntax	:DISPlay:ENABle OFF ON 0 1 :DISPlay:ENABle?
Function	Enables or disables the screen display. The refresh will be stopped when screen display is disabled, pressing ESC to unlock. The query returns "0" or "1".
Default	ON 1
Front Panel	Display, Scr State, On/Off

<b>5. :DISPlay:MENU:STATE</b>	
Syntax	:DISPlay:MENU:STATE OFF ON 0 1 :DISPlay:MENU:STATE?
Function	Turns on or off the menu hold function. The query returns "0" or "1".
Default	ON 1
<b>6. :DISPlay:MENU:HTIME</b>	
Syntax	:DISPlay:MENU:HTIME 3s 5s 10s 20s INFinite :DISPlay:MENU:HTIME?
Function	Sets the menu hold time. The query returns "3S", "5S", "10S", "20S" or "INF".
Default	INFinite
Front Panel	Display, Menu Hold, 3s/5s/10s/20s/Infinite
<b>7. :DISPlay:MSGswitch:STATE</b>	
Syntax	:DISPlay:MSGswitch:STATE OFF ON 0 1 :DISPlay:MSGswitch:STATE?
Function	Enables or disables the message display. The query returns "0" or "1".
Default	ON 1
Front Panel	Display, Msg Switch, On/Off
<b>8. :DISPlay:SKIN</b>	
Syntax	:DISPlay:SKIN CLASsic MODErn :DISPlay:SKIN?
Function	Specifies the screen skin. The query returns "CLAS" or "MODE".
Default	CLASsic
Front Panel	Display, Skin, Classic/Modern
<b>9. :DISPlay:SSAVer:TIME</b>	
Syntax	:DISPlay:SSAVer:TIME 1m 15m 30m 1h OFF :DISPlay:SSAVer:TIME?
Function	Sets the screen saver duration. The query returns "1M", "15M", "30M", "1H" or "OFF".
Default	OFF

Front Panel	Display, Scr Saver, 1min/15mins/30mins /1hour/Off
<b>10. :DISPlay:WINDow:TRACe:GRATicule:GRID</b>	
Syntax	:DISPlay:WINDow:TRACe:GRATicule:GRID <integer> :DISPlay:WINDow:TRACe:GRATicule:GRID?
Function	Sets the grid brightness, such as :DISP:WIN:TRAC:GRAT:GRID 6 The query returns "6".
Explanations	<integer> ranges from 0 to 10.
Default	5
Front Panel	Display, Graticule
<b>11. :DISPlay:WINDow:TRACe:Y:DLINe</b>	
Syntax	:DISPlay:WINDow:TRACe:Y:DLINe <ampl> :DISPlay:WINDow:TRACe:Y:DLINe?
Function	Sets the screen display line, the default unit is dBm, such as :DISP:WIN:TRAC:Y:DLIN -10 The query returns "-1.000000E+01".
Explanations	<ampl> ranges from -100 dBm to 30 dBm.
Default	0 dBm
Front Panel	Display, Display Line
<b>12. :DISPlay:WINDow:TRACe:Y:DLINe:STATe</b>	
Syntax	:DISPlay:WINDow:TRACe:Y:DLINe:STATe OFF ON 0 1 :DISPlay:WINDow:TRACe:Y:DLINe:STATe?
Function	Enables or disables the display line. The query returns "0" or "1".
Default	OFF 0
Front Panel	Display, Display Line, On/Off
<b>13. :DISPlay:WINDow:TRACe:Y:SCALe:PDIVision</b>	
Syntax	:DISPlay:WINDow:TRACe:Y:SCALe:PDIVision <rel_ampl> :DISPlay:WINDow:TRACe:Y:SCALe:PDIVision?
Function	Specifies the scale magnitude of Y-axis, such as :DISP:WIN:TRAC:Y:SCAL:PDIV 10 The query returns "1.000000E+01".
Explanations	<rel_ampl> ranges from 0.1 dB to 20.0 dB.

Default	10 dB
Front Panel	AMPT, Scale/Div
<b>14. :DISPlay:WINDow:TRACe:Y:SCALE:RLEVel</b>	
Syntax	:DISPlay:WINDow:TRACe:Y:SCALE:RLEVel <ampl> :DISPlay:WINDow:TRACe:Y:SCALE:RLEVel?
Function	Sets the reference level, such as :DISP:WIN:TRAC:Y:SCAL:RLEV -10 The query returns "-1.000000E+01".
Explanations	<ul style="list-style-type: none"> <li>• &lt;ampl&gt; ranges from -100 dBm to 30 dBm.</li> <li>• The command may have an influence on the attenuator.</li> </ul>
Default	0 dBm
Front Panel	AMPT, Ref Level
<b>15. :DISPlay:WINDow:TRACe:Y:SCALE:RLEVel:OFFSet</b>	
Syntax	:DISPlay:WINDow:TRACe:Y:SCALE:RLEVel:OFFSet <rel_ampl> :DISPlay:WINDow:TRACe:Y:SCALE:RLEVel:OFFSet?
Function	Sets the reference level offset, such as :DISP:WIN:TRAC:Y:SCAL:RLEV:OFFS 10 The query returns "1.000000E+01".
Explanations	-300 dB to 300 dB
Default	0 dB
Front Panel	AMPT, Ref Offset
<b>16. :DISPlay:WINDow:TRACe:Y:SCALE:SPACing</b>	
Syntax	:DISPlay:WINDow:TRACe:Y:SCALE:SPACing LINear LOGarithmic :DISPlay:WINDow:TRACe:Y:SCALE:SPACing?
Function	Sets the scale type. The query returns "LIN" or "LOG".
Default	LOGarithmic
Front Panel	AMPT, Scale Type, Log/Lin
<b>17. :DISPlay:WINDow:TRACe:Y:SCALE:NRLevel</b>	
Syntax	:DISPlay:WINDow:TRACe:Y:SCALE:NRLevel <rel_ampl> :DISPlay:WINDow:TRACe:Y:SCALE:NRLevel?
Function	Sets the reference level of the normalization, such as :DISP:WIN:TRAC:Y:SCAL:NRL -20

	The query returns "-2.000000E+01".
Explanations	<rel_ampl> ranges from -200 dBm to 200 dBm.
Default	0 dBm
Front Panel	Source, Normalize, Norm Ref Lvl
<b>18. :DISPlay:WINDow:TRACe:Y:SCALe:NRPosition</b>	
Syntax	:DISPlay:WINDow:TRACe:Y:SCALe:NRPosition <integer> :DISPlay:WINDow:TRACe:Y:SCALe:NRPosition?
Function	Sets the reference position of the normalization, such as :DISP:WIN:TRAC:Y:SCAL:NRP 50 The query returns "5.000000E+01".
Explanations	<integer> ranges from 0 to 100.
Default	100
Front Panel	Source, Normalize, Norm Ref Pos

## :FETCh

- :FETCh:ACPower?
- :FETCh:ACPower:LOWer?
- :FETCh:ACPower:UPPer?
- :FETCh:ACPower:MAIN?
- :FETCh:CHPower?
- :FETCh:CHPower:CHPower?
- :FETCh:CHPower:DENSity?
- :FETCh:CNRatio?
- :FETCh:CNRatio:CARRier?
- :FETCh:CNRatio:CNRatio?
- :FETCh:CNRatio:NOISe?
- :FETCh:EBWidth?
- :FETCh:HARMonics:AMPLitude:ALL?
- :FETCh:HARMonics:AMPLitude? <n>
- :FETCh:HARMonics[:DISTortion]?
- :FETCh:HARMonics:FREQuency:ALL?
- :FETCh:HARMonics:FREQuency? <n>
- :FETCh:HARMonics:FUNDamental?
- :FETCh:OBWidth?
- :FETCh:OBWidth:OBWidth?
- :FETCh:OBWidth:OBWidth:FERRor?
- :FETCh:TOIntercept?
- :FETCh:TOIntercept:IP3?
- :FETCh:TPOWer?

**NOTE: The FETCh commands only apply to DSA1030 which has been installed the advanced measurement option.**

<b>1. :FETCh:ACPower?</b>	
Syntax	:FETCh:ACPower?
Function	The query returns 5 values at comma intervals: Main channel power Upper channel power, Power difference between upper and main channel (dBc) Lower channel power, Power difference between lower and main channel (dBc)
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -5.150423E+01, -5.173441E+01, -2.301865E-01, -5.142665E+01,7.757568E-02
<b>2. :FETCh:ACPower:LOWer?</b>	
Syntax	:FETCh:ACPower:LOWer?
Function	The query returns the lower channel power.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -5.142665E+01
<b>3. :FETCh:ACPower:UPPer?</b>	
Syntax	:FETCh:ACPower:UPPer?
Function	The query returns upper channel power.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -5.173441E+01
<b>4. :FETCh:ACPower:MAIN?</b>	
Syntax	:FETCh:ACPower:MAIN?

Function	The query returns the main channel power.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -5.150423E+01
<b>5. :FETCh:CHPower?</b>	
Syntax	:FETCh:CHPower?
Function	The query returns 2 values at comma intervals: channel power and power spectral density.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -1.599480E+01,-7.900511E+01
<b>6. :FETCh:CHPower:CHPower?</b>	
Syntax	:FETCh:CHPower:CHPower?
Function	The query returns the channel power.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -1.599480E+01
<b>7. :FETCh:CHPower:DENSity?</b>	
Syntax	:FETCh:CHPower:DENSity?
Function	The query returns the channel power spectral density.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -7.900511E+01
<b>8. :FETCh:CNRatio?</b>	
Syntax	:FETCh:CNRatio?

Function	The query returns 3 values at comma intervals: carrier power, noise power and carrier-to-noise ratio (dB).
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -6.048788E+01,-6.186192E+01,1.374039E+00
<b>9. :FETCh:CNRatio:CARRier?</b>	
Syntax	:FETCh:CNRatio:CARRier?
Function	The query returns the carrier power.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -1.484203E+01
<b>10. :FETCh:CNRatio:CNRatio?</b>	
Syntax	:FETCh:CNRatio:CNRatio?
Function	The query returns the C/N Ratio in dB.
Explanations	The command transmits the data from the latest measurement to the output buffer.
Return Value	Such as 8.956909E-02
<b>11. :FETCh:CNRatio:NOISe?</b>	
Syntax	:FETCh:CNRatio:NOISe?
Function	The query returns the noise power.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -1.442294E+01
<b>12. :FETCh:EBWidth?</b>	
Syntax	:FETCh:EBWidth?
Function	The query returns the measurement result of emission bandwidth in Hz.

Explanations	The command transmits the data from the latest measurement to the output buffer.
Return Value	Such as 50000
<b>13. :FETCh:HARMonics:AMPLitude:ALL?</b>	
Syntax	:FETCh:HARMonics:AMPLitude:ALL?
Function	The query returns the amplitude value of the first ten harmonics at comma intervals and in the same unit of the current amplitude.
Explanations	<ul style="list-style-type: none"> <li>● If there are less than ten harmonics to measure, the query returns nothing for those unmeasured harmonics.</li> <li>● The unit of the returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -1.692102E+01,-6.458423E+01,-7.509421E+01,-7.924328E+01, -7.847027E+01,-7.885457E+01,-7.882358E+01, -7.921457E+01, -7.923057E+01,-7.915358E+01
<b>14. :FETCh:HARMonics:AMPLitude? &lt;n&gt;</b>	
Syntax	:FETCh:HARMonics:AMPLitude? <n>
Function	The query returns the specified harmonic amplitude, such as :FETC:HARM:AMPL? 1
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is an integer within 1 and 10.</li> <li>● The unit of the returned value is the same as the unit of Y-axis.</li> <li>● The query returns "---" if the harmonic to be read has no data.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -1.692102E+01
<b>15. :FETCh:HARMonics[:DISTortion]?</b>	
Syntax	:FETCh:HARMonics[:DISTortion]?
Function	The query returns the percentage of THD (total harmonic distortion).
Explanations	The command transmits the data from the latest measurement to the output buffer.
Return Value	Such as 2.490393E+02

<b>16. :FETCh:HARMonics:FREQuency:ALL?</b>	
Syntax	:FETCh:HARMonics:FREQuency:ALL?
Function	The query returns the frequency of 10 harmonics at comma intervals in Hz. The first harmonic is the fundamental wave.
Explanations	<ul style="list-style-type: none"> <li>● The command returns "0" for those unmeasured harmonics if less than 10 harmonics to measure.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as 45500000,91000000,136500000,182000000, 227500000, 273000000,318500000,364000000,409500000,455000000
<b>17. :FETCh:HARMonics:FREQuency? &lt;n&gt;</b>	
Syntax	:FETCh:HARMonics:FREQuency? <n>
Function	Queries the frequency of specified harmonic in Hz, such as :FETC:HARM:FREQ? 1
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is an integer within 1 and 10.</li> <li>● The query returns "---" if the harmonic to be read has no data.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as 45500000
<b>18. :FETCh:HARMonics:FUNDamental?</b>	
Syntax	:FETCh:HARMonics:FUNDamental?
Function	The query returns the fundamental wave frequency in Hz.
Explanations	<ul style="list-style-type: none"> <li>● The command is the same as :FETCh:HARMonics:FREQuency? 1</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as 45500000
<b>19. :FETCh:OBWidth?</b>	
Syntax	:FETCh:OBWidth?
Function	The query returns 2 values at comma intervals: Occupied BandWidth (Hz) and Transmit Freq Error (Hz).
Explanations	The command transmits the data from the latest measurement to the output buffer.
Return Value	Such as 1860000,40000

<b>20. :FETCh:OBWidth:OBWidth?</b>	
Syntax	:FETCh:OBWidth:OBWidth?
Function	The query returns the bandwidth of the carrier signal in Hz.
Explanations	The command transmits the data from the latest measurement to the output buffer.
Return Value	Such as 1860000
<b>21. :FETCh:OBWidth:OBWidth:FERRor?</b>	
Syntax	:FETCh:OBWidth:OBWidth:FERRor?
Function	The query returns the frequency transmission error in Hz.
Explanations	The command transmits the data from the latest measurement to the output buffer.
Return Value	Such as 40000
<b>22. :FETCh:TOIntercept?</b>	
Syntax	:FETCh:TOIntercept?
Function	The query returns the following values at comma intervals: Frequency of Base Lower (Hz), Amplitude Frequency of Base Upper (Hz), Amplitude Frequency of 3rd Order Lower (Hz), Amplitude and Intercept of TOI Frequency of 3rd Order Upper (Hz), Amplitude and Intercept of TOI
Explanations	<ul style="list-style-type: none"> <li>● The amplitude unit of the returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as 1500450000,-8.131735E+01, 1500450000,-8.131735E+01, 1500450000,-8.131735E+01,-8.131735E+01, 1500450000,-8.131735E+01,-8.131735E+01
<b>23. :FETCh:TOIntercept:IP3?</b>	
Syntax	:FETCh:TOIntercept:IP3?
Function	The query returns the smaller TOI intercept between the 3rd Order Lower and the 3rd Order Upper.
Explanations	The command transmits the data from the latest measurement to

	the output buffer.
Return Value	Such as -8.131735E+01
<b>24. :FETCh:TPOWer?</b>	
Syntax	:FETCh:TPOWer?
Function	The query returns the Time-domain Power measurement result.
Explanations	<ul style="list-style-type: none"> <li>● The unit of the returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -1.658941E+01

## :FORMat

- :FORMat:BORDER
- :FORMat[:TRACe][:DATA]

<b>1. :FORMat:BORDER</b>	
Syntax	:FORMat:BORDER NORMal SWAPped :FORMat:BORDER?
Function	Specifies the transmission order of the binary data. The query returns "NORM" or "SWAP".
Default	NORMal
<b>2. :FORMat[:TRACe][:DATA]</b>	
Syntax	:FORMat[:TRACe][:DATA] ASCii REAL[,32] :FORMat[:TRACe][:DATA]?
Function	Specifies the input/output format of a trace data. The query returns "ASCII" or "REAL,32".
Default	ASCii

## **:HCOPY**

- :HCOPY:ABORt
- :HCOPY:IMAGe:COLor[:STATe]
- :HCOPY:IMAGe:INVert
- :HCOPY:IMAGe:PTIMe
- :HCOPY:IMAGe:QUALity
- :HCOPY:IMAGe:FTYPe
- :HCOPY[:IMMEDIATE]
- :HCOPY:PAGE:ORientation
- :HCOPY:PAGE:PRINTs
- :HCOPY:PAGE:SIZE
- :HCOPY:RESume

<b>1. :HCOPY:ABORT</b>	
Syntax	:HCOPY:ABORT
Function	Aborts the printing job.
Front Panel	Print Setup, Cancel
<b>2. :HCOPY:IMAGE:COLOR[:STATe]</b>	
Syntax	:HCOPY:IMAGE:COLOR[:STATe] OFF ON 0 1 :HCOPY:IMAGE:COLOR[:STATe]?
Function	Sets the print color to Grey (OFF 0) or Color (ON 1). The query returns "0" or "1".
Explanations	This setting is persistent even through a power cycle.
Default	OFF 0 (Grey)
Front Panel	Print Setup, Palette, Gray/Color
<b>3. :HCOPY:IMAGE:INVERT</b>	
Syntax	:HCOPY:IMAGE:INVERT OFF ON 0 1 :HCOPY:IMAGE:INVERT?
Function	Sets if to use inverse print. The query returns "0" or "1".
Default	OFF 0
Front Panel	Print Setup, Inverted, On/Off
<b>4. :HCOPY:IMAGE:PTIME</b>	
Syntax	:HCOPY:IMAGE:PTIME OFF ON 0 1 :HCOPY:IMAGE:PTIME?
Function	Sets if to print with date. The query returns "0" or "1".
Default	OFF 0
Front Panel	Print Setup, Date Prints, On/Off
<b>5. :HCOPY:IMAGE:QUALITY</b>	
Syntax	:HCOPY:IMAGE:QUALITY DEFault NORMal DRAFt FINE :HCOPY:IMAGE:QUALITY?
Function	Specifies the print quality. The query returns "DEF", "NORM", "DRAF" or "FINE".
Default	DEFault (depends on the printer being used)
Front Panel	Print Setup, Qualities, Default/Normal/Draft/Fine

<b>6. :HCOPY:IMAGe:FTYPE</b>	
Syntax	:HCOPY:IMAGe:FTYPE DEFault EXIFjpeg :HCOPY:IMAGe:FTYPE?
Function	Specifies the image type you want to print. The query returns "DEF" or "EXIF".
Default	DEFault (depends on the printer being used)
Front Panel	Print Setup, File Type, Default or Exif/JPEG
<b>7. :HCOPY[:IMMediate]</b>	
Syntax	:HCOPY[:IMMediate]
Function	Performs the print job.
Front Panel	Print Setup, Print
<b>8. :HCOPY:PAGE:ORientation</b>	
Syntax	:HCOPY:PAGE:ORientation LANDscape PORTrait :HCOPY:PAGE:ORientation?
Function	Specifies the paper orientation you want to print. The query returns "LAND" or "PORT".
Default	LANDscape
Front Panel	Print Setup, Orientation, Portr/Portr
<b>9. :HCOPY:PAGE:PRINts</b>	
Syntax	:HCOPY:PAGE:PRINts <integer> :HCOPY:PAGE:PRINts?
Function	Specifies the number of copies you want to print, such as :HCOP:PAGE:PRIN 10 The query returns "10".
Explanations	<ul style="list-style-type: none"> <li>● &lt;integer&gt; ranges from 1 to 999.</li> <li>● This setting is persistent even through a power cycle.</li> </ul>
Default	1
Front Panel	Print Setup, Copies
<b>10. :HCOPY:PAGE:SIZE</b>	
Syntax	:HCOPY:PAGE:SIZE DEFault A4 A5 A6 B5 :HCOPY:PAGE:SIZE?
Function	Selects a page size from Default, A4, A5, A6 or B5.

	The query returns "DEF", "A4", "A5", "A6" or "B5".
Default	DEfault (depends on the printer being used)
Front Panel	Print Setup, Page Size, Default/A4/A5/A6/B5
<b>11. :HCOPY:RESume</b>	
Syntax	:HCOPY:RESume
Function	Resumes the suspended print job.
Front Panel	Print Setup, Resume

## :INITiate

- :INITiate:CONTinuous
- :INITiate[:IMMEDIATE]
- :INITiate:PAUSE\*
- :INITiate:REStart\*
- :INITiate:RESume\*

**NOTE:** The commands marked with "\*" only apply to DSA1030 which has been installed the advanced measurement option.

<b>1. :INITiate:CONTInuous</b>	
Syntax	:INITiate:CONTInuous OFF ON 0 1 :INITiate:CONTInuous?
Function	Selects Continue (ON 1) or Single (OFF 0) sweep when measurement is not in progress. Selects Continue (ON 1) or Single (OFF 0) measurement when measurement is in progress. The query returns "0" or "1".
Default	ON 1
Front Panel	Sweep, Mode, Single/Cont Meas, Meas Fctn, any Meas Fctn, Meas Mode, Single/Cont
<b>2. :INITiate[:IMMediate]</b>	
Syntax	:INITiate[:IMMediate]
Function	Initializes a sweep when measurement is not in progress. Triggers a measurement when measurement is in progress.
Explanations	<ul style="list-style-type: none"> <li>• The analyzer must be in Single measurement mode.</li> <li>• The command will be ignored if "ON" is specified by command :INITiate:CONTInuous.</li> <li>• Sending :FETCh? command transmits measurement results from the internal memory to the output buffer.</li> </ul>
Front Panel	Single
<b>3. :INITiate:PAUSE</b>	
Syntax	:INITiate:PAUSE
Function	Switches the instrument from "Wait For Trigger" to "Pause".
Explanations	The command is valid only when a measurement is open.
Front Panel	Meas, Off
<b>4. :INITiate:REStart</b>	
Syntax	:INITiate:REStart
Function	Restarts the current measurement in "Idle" state.
Explanations	The command is valid only when a measurement is open.
Front Panel	Meas, Restart
<b>5. :INITiate:RESume</b>	
Syntax	:INITiate:RESume

---

Function	Switches the instrument from "Pause" to "Wait For Trigger".
Explanations	The command is valid only when a measurement is open.
Front Panel	Meas, Resume

## :INPut

- :INPut:IMPedance

1. :INPut:IMPedance	
Syntax	:INPut:IMPedance 50 75 :INPut:IMPedance?
Function	Sets the input impedance for voltage-to-power conversions.
Explanations	To measure a 75Ω device, you should use a 75Ω to 50Ω adapter supplied by <b>RIGOL</b> to connect the analyzer with the system under test and then set the corresponding menu on the front panel to 75Ω.
Default	50Ω
Front Panel	AMPT, Input, 50Ω/75Ω

## :MMEMory

- :MMEMory:DELeTe
- :MMEMory:DISK:FORMat
- :MMEMory:DISK:INFormation?
- :MMEMory:MOVE
- :MMEMory:LOAD:LIMit\*
- :MMEMory:LOAD:MTABle
- :MMEMory:LOAD:CORRection
- :MMEMory:LOAD:STATe
- :MMEMory:LOAD:TRACe
- :MMEMory:LOAD:SETUp
- :MMEMory:STORE:CORRection
- :MMEMory:STORE:SCReen
- :MMEMory:STORE:STATe
- :MMEMory:STORE:TRACe
- :MMEMory:STORE:SETUp
- :MMEMory:STORE:RESults\*
- :MMEMory:STORE:MTABle
- :MMEMory:STORE:PTABle
- :MMEMory:STORE:LIMit\*

**NOTE: The commands marked with "\*" only apply to DSA1030 which has been installed the advanced measurement option.**

<b>1. :MMEMory:DISK:FORMat</b>	
Syntax	:MMEMory:DISK:FORMat
Function	Formats the local disk.
Front Panel	Storage, Disk Mgr, Format (D:)
<b>2. :MMEMory:DISK:INFormation?</b>	
Syntax	:MMEMory:DISK:INFormation?
Function	The query returns the current disk name, disk type, file system, used space and total capacity.
Front Panel	Storage, Disk Mgr, Disk Info
<b>3. :MMEMory:MOVE</b>	
Syntax	:MMEMory:MOVE <file_name1>,<file_name2>
Function	Renames the file that specified by <file_name1> to <file_name2>.
Explanations	<ul style="list-style-type: none"> <li>● &lt;file_name1&gt; and &lt;file_name2&gt; must contain a valid path.</li> <li>● The command is invalid if the file you specify does not exist.</li> </ul>
Example	:MMEM:MOVE D:\1.csv, D:\2.csv
Front Panel	Storage, Rename
<b>4. :MMEMory:LOAD:LIMit</b>	
Syntax	:MMEMory:LOAD:LIMit <file_name>
Function	Loads the edited limit line file (.lim) into the analyzer.
Explanations	<ul style="list-style-type: none"> <li>● &lt;file_name&gt; must contain a valid path.</li> <li>● The command is invalid if the file you specify does not exist.</li> </ul>
Example	:MMEM:LOAD:LIM D:\edit.lim
Front Panel	Storage, File Type, Trace, Recall
<b>5. :MMEMory:LOAD:MTABLE</b>	
Syntax	:MMEMory:LOAD:MTABLE <file_name>
Function	Loads the stored marker file (.mkr or .csv) into the analyzer.
Explanations	<ul style="list-style-type: none"> <li>● &lt;file_name&gt; must contain a valid path.</li> <li>● The command is invalid if the file you specify does not exist.</li> </ul>
Example	:MMEM:LOAD:MTAB E:\table.csv
Front Panel	Storage, File Type, Mark Table, Recall
<b>6. :MMEMory:DELeTe</b>	
Syntax	:MMEMory:DELeTe <file_name>

Function	Deletes the specified file.
Explanations	<ul style="list-style-type: none"> <li>● &lt;file_name&gt; must contain a valid path.</li> <li>● The command is invalid if the file you specify does not exist.</li> </ul>
Example	:MMEM:DEL E:\measure.csv
Front Panel	Storage, Delete
<b>7. :MMEMory:LOAD:CORRection</b>	
Syntax	:MMEMory:LOAD:CORRection ANTenna CABLE OTHer USER,<file_name>
Function	Loads the data from specified file (.cbl) to make an amplitude correction.
Explanations	<ul style="list-style-type: none"> <li>● &lt;file_name&gt; must contain a valid path.</li> <li>● The command is invalid if the file you specify does not exist.</li> </ul>
Example	:MMEM:LOAD:CORR ANT, D:\1.cbl
Front Panel	Storage, File Type, Corrections, Recall
<b>8. :MMEMory:LOAD:STATe</b>	
Syntax	:MMEMory:LOAD:STATe 1,<file_name>
Function	Loads the specified status file (.sta) into the analyzer.
Explanations	<ul style="list-style-type: none"> <li>● &lt;file_name&gt; must contain a valid path.</li> <li>● The command is invalid if the file you specify does not exist.</li> </ul>
Example	:MMEM:LOAD:STAT 1,D:\state.sta
Front Panel	Storage, File Type, Status, Recall
<b>9. :MMEMory:LOAD:TRACe</b>	
Syntax	:MMEMory:LOAD:TRACe <file_name>
Function	Loads the specified file (.trc) into the trace.
Explanations	<ul style="list-style-type: none"> <li>● &lt;file_name&gt; must contain a valid path.</li> <li>● The command is invalid if the file you specify does not exist.</li> </ul>
Example	:MMEM:LOAD:TRAC D:\trace.trc
Front Panel	Storage, File Type, Trace, Recall
<b>10. :MMEMory:LOAD:SETUp</b>	
Syntax	:MMEMory:LOAD:SETUp <file_name>
Function	Loads the specified setting file (.set).
Explanations	<ul style="list-style-type: none"> <li>● &lt;file_name&gt; must contain a valid path.</li> <li>● The command is invalid if the file you specify does not exist.</li> </ul>

Example	:MMEM:LOAD:SETU D:\sys.set
Front Panel	Storage, File Type, Setup, Recall
<b>11. :MMEMory:STORe:CORRection</b>	
Syntax	:MMEMory:STORe:CORRection ANTenna CABLe OTHer USER,<file_name>
Function	Saves the amplitude corrections with a specified file name (.cbl).
Explanations	<ul style="list-style-type: none"> <li>• &lt;file_name&gt; must contain a valid path.</li> <li>• If the file name specified already exists, it will be overwritten.</li> </ul>
Example	:MMEM:STOR:CORR ANT,D:\ANT.cbl
Front Panel	Storage, File Type, Corrections, Save
<b>12. :MMEMory:STORe:SCReen</b>	
Syntax	:MMEMory:STORe:SCReen <file_name>
Function	Saves the current screen into the local disk or a USB flash device with a specified file name (.bmp).
Explanations	<ul style="list-style-type: none"> <li>• &lt;file_name&gt; must contain a valid path.</li> <li>• If the file name specified already exists, it will be overwritten.</li> </ul>
Example	:MMEM:STOR:SCR E:\screen.bmp
Front Panel	Print
<b>13. :MMEMory:STORe:STATe</b>	
Syntax	:MMEMory:STORe:STATe 1,<file_name>
Function	Saves the current status into the internal memory of the analyzer with specified file name (.sta).
Explanations	<ul style="list-style-type: none"> <li>• &lt;file_name&gt; must contain a valid path.</li> <li>• If the file name specified already exists, it will be overwritten.</li> <li>• This file can only be read by the analyzer.</li> </ul>
Example	:MMEM:STOR:STAT 1,D:\state.sta
Front Panel	Storage, File Type, State, Save
<b>14. :MMEMory:STORe:TRACe</b>	
Syntax	:MMEMory:STORe:TRACe <label>,<file_name>
Function	Specifies the trace with a specified file name (.trc).
Explanations	<ul style="list-style-type: none"> <li>• &lt;file_name&gt; must contain a valid path.</li> <li>• &lt;label&gt; can be: TRACE1 TRACE2 TRACE3 ALL.</li> <li>• If the file name specified already exists, it will be overwritten.</li> </ul>

	<ul style="list-style-type: none"> <li>This file can only be read by the analyzer.</li> </ul>
Example	:MMEM:STOR:TRAC TRACE3,D:\trace.trc
Front Panel	Storage, File Type, Trace, Save
<b>15. :MMEMory:STORe:SETUp</b>	
Syntax	:MMEMory:STORe:SETUp <file_name>
Function	Saves the current settings by specified file (.set).
Explanations	<ul style="list-style-type: none"> <li>&lt;file_name&gt; must contain a valid path.</li> <li>If the file name specified already exists, it will be overwritten.</li> </ul>
Example	:MMEM:STOR:SETU D:\state.set
Front Panel	Storage, File Type, Setup, Save
<b>16. :MMEMory:STORe:RESults</b>	
Syntax	:MMEMory:STORe:RESults <file_name>
Function	Saves the current measurement results into a USB flash device with a specified file name (.csv).
Explanations	<ul style="list-style-type: none"> <li>&lt;file_name&gt; must contain a valid path.</li> <li>If the file name specified already exists, it will be overwritten.</li> </ul>
Example	:MMEM:STOR:RES E:\ACP.csv
Front Panel	Storage, File Type, Measure, Save
<b>17. :MMEMory:STORe:MTABLE</b>	
Syntax	:MMEMory:STORe:MTABLE <file_name>
Function	Saves the marker table into a USB flash device with a specified file name (.csv).
Explanations	<ul style="list-style-type: none"> <li>&lt;file_name&gt; must contain a valid path.</li> <li>If the file name specified already exists, it will be overwritten.</li> </ul>
Example	:MMEM:STOR:MTAB E:\marktable.csv
Front Panel	Storage, File Type, Mark Table, Save
<b>18. :MMEMory:STORe:PTABLE</b>	
Syntax	:MMEMory:STORe:PTABLE <file_name>
Function	Saves the peak table into a USB flash device with a specified file name (.csv).
Explanations	<ul style="list-style-type: none"> <li>&lt;file_name&gt; must contain a valid path.</li> <li>If the file name specified already exists, it will be overwritten.</li> </ul>
Example	:MMEM:STOR:PTAB E:\peaktable.csv

Front Panel	Storage, File Type, Peak Table, Save
<b>19. :MMEMory:STORe:LIMit</b>	
Syntax	:MMEMory:STORe:LIMit <file_name>
Function	Saves the current edited limit line with a specified file name (.trc or .csv).
Explanations	<ul style="list-style-type: none"> <li>● &lt;file_name&gt; must contain a valid path.</li> <li>● If the file name specified already exists, it will be overwritten.</li> </ul>
Example	:MMEM:STOR:LIM E:\limit.csv

## :OUTPut

- :OUTPut[:STATe]

<b>1. :OUTPut[:STATe]</b>	
Syntax	:OUTPut[:STATe] OFF ON 0 1 :OUTPut[:STATe]?
Function	Enables or disables the tracking generator output. The query returns "0" or "1".
Explanations	This command only applies to DSA1030 which has been installed the tracking generator option.
Default	OFF 0
Front Panel	Source, TG, On/Off

## :READ

In the command system, both Read and Fetch command can be used to get measurement results. The difference is that a Fetch command can get measurement results immediately and a Read command always starts a measurement and then read the results.

- :READ:ACPower?
- :READ:ACPower:LOWer?
- :READ:ACPower:UPPer?
- :READ:ACPower:MAIN?
- :READ:CHPower?
- :READ:CHPower:CHPower?
- :READ:CHPower:DENSity?
- :READ:CNRatio?
- :READ:CNRatio:CARRier?
- :READ:CNRatio:CNRatio?
- :READ:CNRatio:NOISe?
- :READ:EBWidth?
- :READ:HARMonics:AMPLitude:ALL?
- :READ:HARMonics:AMPLitude? <n>
- :READ:HARMonics[:DISTortion]?
- :READ:HARMonics:FREQuency:ALL?
- :READ:HARMonics:FREQuency? <n>
- :READ:HARMonics:FUNDamental?
- :READ:OBWidth?
- :READ:OBWidth:OBWidth?
- :READ:OBWidth:OBWidth:FERRor?
- :READ:TOIntercept?
- :READ:TOIntercept:IP3?
- :READ:TPOWER?

**NOTE: The READ commands only apply to DSA1030 which has been installed the advanced measurement option.**

<b>1. :READ:ACPower?</b>	
Syntax	:READ:ACPower?
Function	The command executes a measurement and returns 5 values at comma intervals: Main channel power Upper channel power, Power difference between upper and main channel (dBc) Lower channel power, Power difference between lower and main channel (dBc)
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	such as -5.150423E+01, -5.173441E+01, -2.301865E-01, -5.142665E+01,7.757568E-02
<b>2. :READ:ACPower:LOWer?</b>	
Syntax	:READ:ACPower:LOWer?
Function	The command executes a measurement and returns the lower channel power.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -5.142665E+01
<b>3. :READ:ACPower:UPPer?</b>	
Syntax	:READ:ACPower:UPPer?
Function	The command executes a measurement and returns the upper channel power.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -5.173441E+01

<b>4. :READ:ACPower:MAIN?</b>	
Syntax	:READ:ACPower:MAIN?
Function	The command executes a measurement and returns the main channel power.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -5.150423E+01
<b>5. :READ:CHPower?</b>	
Syntax	:READ:CHPower?
Function	The command executes a measurement and returns 2 values at comma intervals: the channel power and the power spectral density.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -1.599480E+01,-7.900511E+01
<b>6. :READ:CHPower:CHPower?</b>	
Syntax	:READ:CHPower:CHPower?
Function	The command executes a measurement and returns the channel power.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -1.599480E+01
<b>7. :READ:CHPower:DENSity?</b>	
Syntax	:READ:CHPower:DENSity?
Function	The command executes a measurement and returns the power spectral density.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of</li> </ul>

	<p>Y-axis.</p> <ul style="list-style-type: none"> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -7.900511E+01
<b>8. :READ:CNRatio?</b>	
Syntax	:READ:CNRatio?
Function	The command executes a measurement and returns 3 values at comma intervals: the carrier power, the noise power and carrier-to-noise ratio (dB).
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -6.048788E+01,-6.186192E+01,1.374039E+00
<b>9. :READ:CNRatio:CARRier?</b>	
Syntax	:READ:CNRatio:CARRier?
Function	The command executes a measurement and returns the carrier power.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -1.484203E+01
<b>10. :READ:CNRatio:CNRatio?</b>	
Syntax	:READ:CNRatio:CNRatio?
Function	The command executes a measurement and returns the noise ratio in dB.
Explanations	The command transmits the data from the latest measurement to the output buffer.
Return Value	Such as 8.956909E-02
<b>11. :READ:CNRatio:NOISe?</b>	
Syntax	:READ:CNRatio:NOISe?
Function	The command executes a measurement and returns the noise

	power.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -1.442294E+01
<b>12. :READ:EBWidth?</b>	
Syntax	:READ:EBWidth?
Function	The command executes a measurement and returns the measurement result of emission bandwidth in Hz.
Explanations	The command transmits the data from the latest measurement to the output buffer.
Return Value	Such as 50000
<b>13. :READ:HARMonics:AMPLitude:ALL?</b>	
Syntax	:READ:HARMonics:AMPLitude:ALL?
Function	The command executes a measurement and returns the amplitude value of the first ten harmonics at comma intervals and having the same unit as the current amplitude.
Explanations	<ul style="list-style-type: none"> <li>● If there are less than ten harmonics to measure, the query returns nothing for those unmeasured harmonics.</li> <li>● The unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -1.692102E+01,-6.458423E+01,-7.509421E+01,-7.924328E+01, -7.847027E+01,-7.885457E+01,-7.882358E+01, -7.921457E+01, -7.923057E+01,-7.915358E+01
<b>14. :READ:HARMonics:AMPLitude? &lt;n&gt;</b>	
Syntax	:READ:HARMonics:AMPLitude? <n>
Function	The command executes a measurement and returns the specified harmonic amplitude, such as :READ:HARM:AMPL? 1
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is an integer within 1 and 10.</li> <li>● The unit of returned value is the same as the unit of Y-axis.</li> </ul>

	<ul style="list-style-type: none"> <li>● The query returns "---" if the harmonic to be read has no data.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -1.692102E+01
<b>15. :READ:HARMonics[:DISTortion]?</b>	
Syntax	:READ:HARMonics[:DISTortion]?
Function	The command executes a measurement and returns the percentage of THD (total harmonic distortion).
Explanations	The command transmits the data from the latest measurement to the output buffer.
Return Value	Such as 2.490393E+02
<b>16. :READ:HARMonics:FREQuency:ALL?</b>	
Syntax	:READ:HARMonics:FREQuency:ALL?
Function	The command executes a measurement and returns the first ten harmonic frequency values at comma intervals and in Hz. The first harmonic is the fundamental wave.
Explanations	<ul style="list-style-type: none"> <li>● The command returns "0" for those unmeasured harmonics if less than 10 harmonics to measure.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as 45500000,91000000,136500000,182000000, 227500000, 273000000,318500000,364000000,409500000,455000000
<b>17. :READ:HARMonics:FREQuency? &lt;n&gt;</b>	
Syntax	:READ:HARMonics:FREQuency? <n>
Function	The command executes a measurement and returns the specified harmonic frequency in Hz, such as :READ:HARM:FREQ? 1
Explanations	<ul style="list-style-type: none"> <li>● &lt;n&gt; is an integer within 1 and 10.</li> <li>● The query returns "---" if the harmonic to be read has no data.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as 45500000
<b>18. :READ:HARMonics:FUNDamental?</b>	

Syntax	:READ:HARMonics:FUNDamental?
Function	The command executes a measurement and returns the fundamental wave frequency in Hz.
Explanations	<ul style="list-style-type: none"> <li>● The command is the same as :READ:HARMonics:FREQuency? 1</li> <li>● The command transmits the data from the latest measurement to the ouput buffer.</li> </ul>
Return Value	Such as 45500000
<b>19. :READ:OBWidth?</b>	
Syntax	:READ:OBWidth?
Function	The command executes a measurement and returns 2 values at comma intervals: the occupied bandwidth (Hz) and the ransmit frequency error (Hz).
Explanations	The command transmits the data from the latest measurement to the ouput buffer.
Return Value	Such as 1860000,40000
<b>20. :READ:OBWidth:OBWidth?</b>	
Syntax	:READ:OBWidth:OBWidth?
Function	The command executes a measurement and returns the carrier signal bandwidth in Hz.
Explanations	The command transmits the data from the latest measurement to the ouput buffer.
Return Value	Such as 1860000
<b>21. :READ:OBWidth:OBWidth:FERRor?</b>	
Syntax	:READ:OBWidth:OBWidth:FERRor?
Function	The command executes a measurement and returns the transmit frequency error in Hz.
Explanations	The command transmits the data from the latest measurement to the ouput buffer.
Return Value	Such as 40000
<b>22. :READ:TOIntercept?</b>	
Syntax	:READ:TOIntercept?
Function	The command executes a measurement and returns the following

	values at comma intervals: Frequency of Base Lower (Hz), Amplitude Frequency of Base Upper (Hz), Amplitude Frequency of 3rd Order Lower (Hz), Amplitude and Intercept of TOI Frequency of 3rd Order Upper (Hz), Amplitude and Intercept of TOI
Explanations	<ul style="list-style-type: none"> <li>● The amplitude unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as 1500450000,-8.131735E+01, 1500450000,-8.131735E+01, 1500450000,-8.131735E+01,-8.131735E+01, 1500450000,-8.131735E+01,-8.131735E+01
<b>23. :READ:TOIntercept:IP3?</b>	
Syntax	:READ:TOIntercept:IP3?
Function	The command executes a measurement and returns the smaller of the TOI intercept between the 3rd Order Lower and the 3rd Order Upper.
Explanations	The command transmits the data from the latest measurement to the output buffer.
Return Value	Such as -8.131735E+01
<b>24. :READ:TPOWer?</b>	
Syntax	:READ:TPOWer?
Function	The command executes a measurement and returns the Time-domain power measurement result.
Explanations	<ul style="list-style-type: none"> <li>● The power unit of returned value is the same as the unit of Y-axis.</li> <li>● The command transmits the data from the latest measurement to the output buffer.</li> </ul>
Return Value	Such as -1.658941E+01

## [**:SENSe**]

- [**:SENSe**]:BANDwidth:RESolution
- [**:SENSe**]:BANDwidth:RESolution:AUTO
- [**:SENSe**]:BANDwidth:VIDeo
- [**:SENSe**]:BANDwidth:VIDeo:AUTO
- [**:SENSe**]:BANDwidth:VIDeo:RATio
- [**:SENSe**]:CORRection:CSET:ALL:DELeTe
- [**:SENSe**]:CORRection:CSET:ALL[:STATe]
- [**:SENSe**]:CORRection:CSET<n>:DATA
- [**:SENSe**]:CORRection:CSET<n>:DATA:MERGe
- [**:SENSe**]:CORRection:CSET<n>:DELeTe
- [**:SENSe**]:CORRection:CSET<n>:X:SPACing
- [**:SENSe**]:CORRection:CSET<n>[:STATe]
- [**:SENSe**]:CORRection:CSET:TABLE:STATe
- [**:SENSe**]:DEMod
- [**:SENSe**]:DEMod:STATe
- [**:SENSe**]:DEMod:TIME
- [**:SENSe**]:DETEctor[:FUNCTion]
- [**:SENSe**]:FREQUency:CENTer
- [**:SENSe**]:FREQUency:CENTer:SET:STEP
- [**:SENSe**]:FREQUency:CENTer:STEP:AUTO
- [**:SENSe**]:FREQUency:CENTer:STEP[:INCRement]
- [**:SENSe**]:FREQUency:CENTer:UP
- [**:SENSe**]:FREQUency:CENTer:DOWN
- [**:SENSe**]:FREQUency:SPAN
- [**:SENSe**]:FREQUency:SPAN:FULL
- [**:SENSe**]:FREQUency:SPAN:PREVious
- [**:SENSe**]:FREQUency:SPAN:ZIN
- [**:SENSe**]:FREQUency:SPAN:ZOUT
- [**:SENSe**]:FREQUency:STARt
- [**:SENSe**]:FREQUency:STOP
- [**:SENSe**]:POWer[:RF]:ATTenuation
- [**:SENSe**]:POWer[:RF]:ATTenuation:AUTO
- [**:SENSe**]:POWer[:RF]:GAIN[:STATe]\*\*
- [**:SENSe**]:POWer[:RF]:MIXer:RANGe[:UPPer]
- [**:SENSe**]:POWer:ASCale

- [:SENSe]:POWer:ARANge
- [:SENSe]:POWer:ATUNe
- [:SENSe]:SWEep:POINts
- [:SENSe]:SWEep:COUNT
- [:SENSe]:SWEep:COUNT:CURRent?
- [:SENSe]:SWEep:TIME
- [:SENSe]:SWEep:TIME:AUTO
- [:SENSe]:SWEep:TIME:AUTO:RULEs
- [:SENSe]:ACPower:AVERage:COUNT\*
- [:SENSe]:ACPower:AVERage[:STATe]\*
- [:SENSe]:ACPower:AVERage:TCONtrol\*
- [:SENSe]:ACPower:BANDwidth:INtegration\*
- [:SENSe]:ACPower:BANDwidth:ACHannel\*
- [:SENSe]:ACPower:CSPacing\*
- [:SENSe]:CHPower:AVERage:COUNT\*
- [:SENSe]:CHPower:AVERage[:STATe]\*
- [:SENSe]:CHPower:AVERage:TCONtrol\*
- [:SENSe]:CHPower:BANDwidth:INtegration\*
- [:SENSe]:CHPower:FREQuency:SPAN\*
- [:SENSe]:CNRatio:AVERage:COUNT\*
- [:SENSe]:CNRatio:AVERage[:STATe]\*
- [:SENSe]:CNRatio:AVERage:TCONtrol\*
- [:SENSe]:CNRatio:BANDwidth:INtegration\*
- [:SENSe]:CNRatio:BANDwidth:NOISe\*
- [:SENSe]:CNRatio:OFFSet\*
- [:SENSe]:EBWidth:AVERage:COUNT\*
- [:SENSe]:EBWidth:AVERage[:STATe]\*
- [:SENSe]:EBWidth:AVERage:TCONtrol\*
- [:SENSe]:EBWidth:MAXHold:STATe\*
- [:SENSe]:EBWidth:FREQuency:SPAN\*
- [:SENSe]:EBWidth:XDB\*
- [:SENSe]:HDISt:AVERage:COUNT\*
- [:SENSe]:HDISt:AVERage[:STATe]\*
- [:SENSe]:HDISt:AVERage:TCONtrol\*
- [:SENSe]:HDISt:NUMBers\*
- [:SENSe]:HDISt:TIME\*
- [:SENSe]:HDISt:TIME:AUTO[:STATe]\*
- [:SENSe]:OBWidth:AVERage:COUNT\*

- [:SENSe]:OBWidth:AVERAge [:STATe] \*
- [:SENSe]:OBWidth:AVERAge:TCONtrol\*
- [:SENSe]:OBWidth:MAXHold:STATe\*
- [:SENSe]:OBWidth:FREQUency:SPAN\*
- [:SENSe]:OBWidth:PERCent\*
- [:SENSe]:ROSCillator:SOURce
- [:SENSe]:STATus:ANALyzer?
- [:SENSe]:STATus:CORRections?
- [:SENSe]:STATus:DEMod?
- [:SENSe]:STATus:POWER? \*
- [:SENSe]:STATus:TG?\*\*\*
- [:SENSe]:TOI:AVERAge:COUNT\*
- [:SENSe]:TOI:AVERAge[:STATe]\*
- [:SENSe]:TOI:AVERAge:TCONtrol\*
- [:SENSe]:TOI:FREQUency:SPAN\*
- [:SENSe]:TPOWer:AVERAge:COUNT\*
- [:SENSe]:TPOWer:AVERAge[:STATe] \*
- [:SENSe]:TPOWer:AVERAge:TCONtrol\*
- [:SENSe]:TPOWer:MODE\*
- [:SENSe]:TPOWer:LLIMit\*
- [:SENSe]:TPOWer:RLIMit\*

**NOTE:**

**The commands marked with "\*" only apply to DSA1030 which has been installed the advanced measurement option.**

**The commands marked with "\*\*\*" only apply to DSA1030 which has been installed the preamplifier option.**

**The commands marked with "\*\*\*\*" only apply to DSA1030 which has been installed the tracking generator option.**

<b>1. [:SENSe]:BANDwidth:RESolution</b>	
Syntax	[:SENSe]:BANDwidth:RESolution <freq> [:SENSe]:BANDwidth:RESolution?
Function	Sets the resolution bandwidth, the default unit is Hz, such as :BAND:RES 1000 or :BAND:RES 1KHZ The query returns 1000.
Explanations	<freq> ranges from 100 Hz to 1 MHz.
Default	1 MHz
Front Panel	BW, RBW
<b>2. [:SENSe]:BANDwidth:RESolution:AUTO</b>	
Syntax	[:SENSe]:BANDwidth:RESolution:AUTO OFF ON 0 1 [:SENSe]:BANDwidth:RESolution:AUTO?
Function	Enables or disables the Auto function of Resolution bandwidth. The query returns "0" or "1".
Explanations	<ul style="list-style-type: none"> <li>● In Auto mode, the resolution bandwidth decreases with the decreasing span.</li> <li>● The Auto function is invalid in Zero-span mode.</li> </ul>
Default	ON 1
Front Panel	BW, RBW, Auto/Manual
<b>3. [:SENSe]:BANDwidth:VIDeo</b>	
Syntax	[:SENSe]:BANDwidth:VIDeo <freq> [:SENSe]:BANDwidth:VIDeo?
Function	Sets the video bandwidth, the default unit is Hz, such as :BAND:VID 1000 or :BAND:VID 1KHZ The query returns "1000".
Explanations	<freq> ranges from 1 Hz to 3 MHz.
Default	1 MHz
Front Panel	BW, VBW
<b>4. [:SENSe]:BANDwidth:VIDeo:AUTO</b>	
Syntax	[:SENSe]:BANDwidth:VIDeo:AUTO OFF ON 0 1 [:SENSe]:BANDwidth:VIDeo:AUTO?
Function	Enables or disables the Auto function of Video bandwidth. The query returns "0" or "1".
Default	ON 1

Front Panel	BW, VBW, Auto/Manual
<b>5. [:SENSe]:BANDwidth:VIDeo:RATio</b>	
Syntax	[:SENSe]:BANDwidth:VIDeo:RATio <number> [:SENSe]:BANDwidth:VIDeo:RATio?
Function	Sets the ratio of VBW to RBW, such as :BAND:VID:RAT 0.01 The query returns "1.000000E-02".
Explanations	<number> ranges from 0.000001 to 300000.
Default	1
Front Panel	BW, V/R Ratio
<b>6. [:SENSe]:CORRection:CSET:ALL:DELeTe</b>	
Syntax	[:SENSe]:CORRection:CSET:ALL:DELeTe
Function	Deletes all amplitude correction data.
Front Panel	AMPT, Corrections, Delete
<b>7. [:SENSe]:CORRection:CSET:ALL[:STATe]</b>	
Syntax	[:SENSe]:CORRection:CSET:ALL[:STATe] OFF ON 0 1 [:SENSe]:CORRection:CSET:ALL[:STATe]?
Function	Enables or disables the amplitude correction function. Only the opened correction factors can be enabled when the correction function is On. All correction factors are disabled when the correction function is Off. The query returns "0" or "1".
Explanations	To turn on or off a single correction factor, see command below: [:SENSe]:CORRection:CSET<n>[:STATe]
Default	OFF 0
Front Panel	AMPT, Corrections, Select, Off/Antenna/Cable/Other/User
<b>8. [:SENSe]:CORRection:CSET&lt;n&gt;:DATA</b>	
Syntax	[:SENSe]:CORRection:CSET<n>:DATA <freq>,<rel_ampl>{,<freq>,<rel_ampl>} [:SENSe]:CORRection:CSET<n>:DATA?
Function	Creates an amplitude correction curve on the basis of current data, such as

	:CORR:CSET1:DATA 900E6,0.3,1.0E9,0.35,1.3E9,0.2 The query returns "900000000,0.300000,1000000000,0.350000, 1300000000,0.200000".
Explanations	<ul style="list-style-type: none"> <li>• &lt;n&gt; is 1 (Antenna), 2 (Cable), 3 (Other) or 4 (User).</li> <li>• &lt;freq&gt; is related to the current analyzer model, the unit is Hz.</li> <li>• &lt;rel_ampl&gt; ranges from -120 dB to 100 dB.</li> <li>• The editable numbers of points range from 1 to 200.</li> </ul>
Front Panel	AMPT, Corrections, Select, Off/Antenna/Cable/Other/User, Edit, Point/Frequency/Amplitude/Del Point
<b>9. [:SENSe]:CORRection:CSET&lt;n&gt;:DATA:MERGe</b>	
Syntax	[:SENSe]:CORRection:CSET<n>:DATA:MERGe <freq>,<rel_ampl>{,<freq>,<rel_ampl>}
Function	Adds new correction data upon the current correction curve.
Explanations	For detailed parameter information, please refer to [:SENSe]:CORRection:CSET<n>:DATA.
<b>10. [:SENSe]:CORRection:CSET&lt;n&gt;:DELeTe</b>	
Syntax	[:SENSe]:CORRection:CSET<n>:DELeTe
Function	Deletes the correction settings.
Explanations	<n> is 1 (Antenna), 2 (Cable), 3 (Other) or 4 (User).
Front Panel	AMPT, Corrections, Select, Off/Antenna/Cable/Other/User, Delete
<b>11. [:SENSe]:CORRection:CSET&lt;n&gt;:X:SPACing</b>	
Syntax	[:SENSe]:CORRection:CSET<n>:X:SPACing LINear LOGarithmic [:SENSe]:CORRection:CSET<n>:X:SPACing?
Function	Sets the amplitude correction mode to Lin or Log.
Explanations	<ul style="list-style-type: none"> <li>• &lt;n&gt; is 1 (Antenna), 2 (Cable), 3 (Other) or 4 (User).</li> <li>• In Lin mode, the frequency uses linear and the amplitude uses logarithm as the unit to make an interpolation operation.</li> <li>• In Log mode, both the frequency and amplitude use logarithm as the unit to make an interpolation operation.</li> </ul>
Default	LINear
Front Panel	AMPT, Corrections, Freq Interp, Log/ Lin
<b>12. [:SENSe]:CORRection:CSET&lt;n&gt;[:STATe]</b>	

Syntax	[[:SENSe]:CORRection:CSET<n>[:STATe] OFF ON 0 1 [:SENSe]:CORRection:CSET<n>[:STATe]?
Function	Turns on or off the specified amplitude correction. The query returns "0" or "1".
Explanations	<n> is 1 (Antenna), 2 (Cable), 3 (Other) or 4 (User).
Default	OFF 0
Front Panel	AMPT, Corrections, Select, Antenna/Cable/Other/User, Correction, On/Off
<b>13. [:SENSe]:CORRection:CSET:TABLE:STATe</b>	
Syntax	[[:SENSe]:CORRection:CSET:TABLE:STATe OFF ON 0 1 [:SENSe]:CORRection:CSET:TABLE:STATe?
Function	Turns on or off the correction table. The query returns "0" or "1".
Default	OFF 0
Front Panel	AMPT, Corrections, Corr Table, On/Off
<b>14. [:SENSe]:DEMod</b>	
Syntax	[[:SENSe]:DEMod AM FM OFF [:SENSe]:DEMod?
Function	Sets the demodulation type. The query returns "AM", "FM" or "OFF".
Front Panel	Demod, Demod, AM/FM
<b>15. [:SENSe]:DEMod:STATe</b>	
Syntax	[[:SENSe]:DEMod:STATe OFF ON 0 1 [:SENSe]:DEMod:STATe?
Function	Turns on or off the demodulation. The query returns "0" or "1".
Default	OFF 0
Front Panel	Demod, Demod, AM/FM/Off
<b>16. [:SENSe]:DEMod:TIME</b>	
Syntax	[[:SENSe]:DEMod:TIME <time> [:SENSe]:DEMod:TIME?
Function	Sets the demodulating time, the default unit is s, such as DEM:TIME 0.5 or DEM:TIME 500ms

	The query returns "5.000000E-01".
Explanations	<time> ranges from 5 ms to 1000000 s.
Default	100 ms
Front Panel	Demod, Demod, AM/FM, Demod Setup, Demod Time
<b>17. [:SENSe]:DETEctor[:FUNction]</b>	
Syntax	[:SENSe]:DETEctor[:FUNction] NEGative NORMal POSitive RMS SAMPlE VAVerage [:SENSe]:DETEctor[:FUNction]?
Function	Sets the detection type. The query returns "NEG", "NORM", "POS", "RMS", "SAMP" or "VAV".
Default	POSitive
Front Panel	Detector, Pos Peak/Neg Peak/Sample/Normal/RMS Avg/Voltage Avg
<b>18. [:SENSe]:FREQuency:CENTer</b>	
Syntax	[:SENSe]:FREQuency:CENTer <freq> [:SENSe]:FREQuency:CENTer?
Function	Sets the center frequency, the default unit is Hz, such as :FREQ:CENT 1000000 or :FREQ:CENT 1MHZ The query returns "1000000".
Explanations	DSA1030: <freq> ranges from 0 Hz to 3 GHz. DSA1020: <freq> ranges from 0 Hz to 2 GHz.
Default	DSA1030: 1.5 GHz DSA1020: 1 GHz
Front Panel	FREQ, Center Freq
<b>19. [:SENSe]:FREQuency:CENTer:SET:STEP</b>	
Syntax	[:SENSe]:FREQuency:CENTer:SET:STEP
Function	Sets the current center frequency step to the center frequency.
Front Panel	FREQ, CF->Step
<b>20. [:SENSe]:FREQuency:CENTer:STEP:AUTO</b>	
Syntax	[:SENSe]:FREQuency:CENTer:STEP:AUTO OFF ON 0 1 [:SENSe]:FREQuency:CENTer:STEP:AUTO?
Function	Enables or disables the Auto function of CF step. The query returns "0" or "1".
Default	ON 1

Front Panel	FREQ, CF Step, Auto/Manual
<b>21. [:SENSe]:FREQuency:CENTer:STEP[:INCRement]</b>	
Syntax	[:SENSe]:FREQuency:CENTer:STEP[:INCRement] <freq> [:SENSe]:FREQuency:CENTer:STEP[:INCRement]?
Function	Sets the CF step, the default unit is Hz, such as :FREQ:CENT:STEP 100000 or :FREQ:CENT:STEP 100KHZ The query returns "100000".
Explanations	DSA1030: <freq> ranges from 1 Hz to 3 GHz. DSA1020: <freq> ranges from 1 Hz to 2 GHz.
Default	Span/10
Front Panel	FREQ, CF Step
<b>22. [:SENSe]:FREQuency:CENTer:UP</b>	
Syntax	[:SENSe]:FREQuency:CENTer:UP
Function	Increases the center frequency by the specified CF step.
Front Panel	FREQ, Channel, Forward
<b>23. [:SENSe]:FREQuency:CENTer:DOWN</b>	
Syntax	[:SENSe]:FREQuency:CENTer:DOWN
Function	Decreases the center frequency by the specified CF step.
Front Panel	FREQ, Channel, Reverse
<b>24. [:SENSe]:FREQuency:SPAN</b>	
Syntax	[:SENSe]:FREQuency:SPAN <freq> [:SENSe]:FREQuency:SPAN?
Function	Sets the span, the default unit is Hz, such as :FREQ:SPAN 20000000 or :FREQ:SPAN 20MHZ The query returns "20000000".
Explanations	<ul style="list-style-type: none"> <li>● &lt;freq&gt; ranges from: 0 Hz to 3 GHz for DSA1030 0 Hz to 2 GHz for DSA1020</li> <li>● When the span is set to 0, the analyzer goes into the Zero-span mode, the X-axis changes from Frequency into Time and only the signal that has the same frequency as the center frequency would be displayed.</li> </ul>
Default	DSA1030: 3 GHz

	DSA1020: 2 GHz
Front Panel	SPAN, Span
<b>25. [:SENSe]:FREQuency:SPAN:FULL</b>	
Syntax	[:SENSe]:FREQuency:SPAN:FULL
Function	Sets the span to the maximum.
Front Panel	SPAN, Full Span
<b>26. [:SENSe]:FREQuency:SPAN:PREVious</b>	
Syntax	[:SENSe]:FREQuency:SPAN:PREVious
Function	Recalls the last span setting.
Front Panel	SPAN, Last Span
<b>27. [:SENSe]:FREQuency:SPAN:ZIN</b>	
Syntax	[:SENSe]:FREQuency:SPAN:ZIN
Function	Sets the span to the half of its current value.
Front Panel	SPAN, Zoom In
<b>28. [:SENSe]:FREQuency:SPAN:ZOUT</b>	
Syntax	[:SENSe]:FREQuency:SPAN:ZOUT
Function	Sets the span to twice the current value.
Front Panel	SPAN, Zoom Out
<b>29. [:SENSe]:FREQuency:STARt</b>	
Syntax	[:SENSe]:FREQuency:STARt <freq> [:SENSe]:FREQuency:STARt?
Function	Sets the start frequency, the default unit is Hz, such as :FREQ:STAR 10000 or :FREQ:STAR 10KHZ The query returns "10000".
Explanations	DSA1030: <freq> ranges from 0 Hz to 3 GHz. DSA1020: <freq> ranges from 0 Hz to 2 GHz.
Default	0 Hz
Front Panel	FREQ, Start Freq
<b>30. [:SENSe]:FREQuency:STOP</b>	
Syntax	[:SENSe]:FREQuency:STOP <freq> [:SENSe]:FREQuency:STOP?

Function	Sets the stop frequency, the default unit is Hz, such as :FREQ:STOP 100000000 or :FREQ:STOP 100MHZ The query returns "100000000".
Explanations	DSA1030: <freq> ranges from 0 Hz to 3 GHz. DSA1020: <freq> ranges from 0 Hz to 2 GHz.
Default	DSA1030: 3 GHz DSA1020: 2 GHz
Front Panel	FREQ, Stop Freq
<b>31. [:SENSe]:POWer[:RF]:ATTenuation</b>	
Syntax	[:SENSe]:POWer[:RF]:ATTenuation <rel_ampl> [:SENSe]:POWer[:RF]:ATTenuation?
Function	Sets the attenuator of the front RF in dB, such as :POW:ATT 20 The query returns "20".
Explanations	<rel_ampl> ranges from 0 dB to 50 dB.
Default	10 dB
Front Panel	AMPT, Input Atten
<b>32. [:SENSe]:POWer[:RF]:ATTenuation:AUTO</b>	
Syntax	[:SENSe]:POWer[:RF]:ATTenuation:AUTO OFF ON 0 1 [:SENSe]:POWer[:RF]:ATTenuation:AUTO?
Function	Enables or disables the Auto function of the attenuator. The query returns "0" or "1".
Default	ON 1
Front Panel	AMPT, Input Atten, Auto/Manual
<b>33. [:SENSe]:POWer[:RF]:GAIN[:STATE]</b>	
Syntax	[:SENSe]:POWer[:RF]:GAIN[:STATE] OFF ON 0 1 [:SENSe]:POWer[:RF]:GAIN[:STATE]?
Function	Turns on or off the preamplifier. The query returns "0" or "1".
Default	OFF 0
Front Panel	AMPT, RF Preamp, On/Off
<b>34. [:SENSe]:POWer[:RF]:MIXer:RANGe[:UPPer]</b>	
Syntax	[:SENSe]:POWer[:RF]:MIXer:RANGe[:UPPer] <ampl>

	<code>[[:SENSe]:POWer[:RF]:MIXer:RANGe[:UPPer]?</code>
Function	Sets the maximum power of the input mixer, the default unit is dBm, such as <code>:POW:MIX:RANG -20</code> The query returns "-2.000000E+01".
Explanations	<ampl> ranges from -50 dBm to 0 dBm.
Default	-10 dBm
Front Panel	AMPT, MaxMixL
<b>35. [[:SENSe]:POWer:AScale</b>	
Syntax	<code>[[:SENSe]:POWer:AScale</code>
Function	Adjusts the reference level and the scale automatically.
Front Panel	AMPT, Auto Scale
<b>36. [[:SENSe]:POWer:ARANGe</b>	
Syntax	<code>[[:SENSe]:POWer:ARANGe</code>
Function	Adjusts the parameters related to the amplitude within the current span range in order to make signal easier to observe.
Front Panel	AMPT, Auto Range
<b>37. [[:SENSe]:POWer:ATUNE</b>	
Syntax	<code>[[:SENSe]:POWer:ATUNE</code>
Function	Searches signals within full range and adjusts the analyzer to the best frequency and amplitude for displaying these signals.
Front Panel	Auto Tune
<b>38. [[:SENSe]:SWEep:POINts</b>	
Syntax	<code>[[:SENSe]:SWEep:POINts &lt;number of points&gt;</code> <code>[[:SENSe]:SWEep:POINts?</code>
Function	Sets the points obtained in each sweep, such as <code>:SWE:POIN 1000</code> The query returns "1000".
Explanations	<ul style="list-style-type: none"> <li>• &lt;number of points&gt; ranges from 101 to 3001.</li> <li>• The sweep points will automatically change to the maximum (3001) and can not be modified by user in <b>Video</b> trigger type (See <b>:TRIGger:SEquence:SOURce</b>).</li> </ul>
Default	601 (equals to the screen points)

Front Panel	Sweep, Points
<b>39. [:SENSe]:SWEep:COUNT</b>	
Syntax	[:SENSe]:SWEep:COUNT <integer> [:SENSe]:SWEep:COUNT?
Function	Specifies the numbers of sweeps in Single sweep mode, such as :SWE:COUN 10 The query returns "10".
Explanations	<integer> ranges from 1 to 9999.
Default	1
Front Panel	Sweep, Numbers
<b>40. [:SENSe]:SWEep:COUNT:CURRENT?</b>	
Syntax	[:SENSe]:SWEep:COUNT:CURRENT?
Function	The query returns the current number of sweeps in Single sweep mode, such as 8.
<b>41. [:SENSe]:SWEep:TIME</b>	
Syntax	[:SENSe]:SWEep:TIME <time> [:SENSe]:SWEep:TIME?
Function	Sets the sweep time of the analyzer within specified span range, the default unit is s, such as :SWEep:TIME 0.1 The query returns "1.000000E-01".
Explanations	DSA1030: <time> ranges from 20 us to 3000 s. DSA1020: <time> ranges from 20 us to 2000 s.
Default	10 ms
Front Panel	Sweep, Time
<b>42. [:SENSe]:SWEep:TIME:AUTO</b>	
Syntax	[:SENSe]:SWEep:TIME:AUTO OFF ON 0 1 [:SENSe]:SWEep:TIME:AUTO?
Function	Turns on or off the Auto function of the sweep time. The query returns "0" or "1".
Default	ON 1
Front Panel	Sweep, Time, Auto/Manual

<b>43. [:SENSe]:SWEep:TIME:AUTO:RULEs</b>	
Syntax	:SENSe:SWEep:TIME:AUTO:RULEs NORMAl ACCuracy :SENSe:SWEep:TIME:AUTO:RULEs?
Function	Sets the auto mode of the sweep time. The query returns "NORM" or "ACC".
Default	NORMAl
Front Panel	Sweep, Auto SWT
<b>44. [:SENSe]:ACPower:AVERAge:COUNT</b>	
Syntax	[:SENSe]:ACPower:AVERAge:COUNT <integer> [:SENSe]:ACPower:AVERAge:COUNT?
Function	Specifies the number of averages of the adjacent channel power measurement results, such as :ACP:AVER:COUN 100 The query returns "100".
Explanations	<integer> ranges from 1 to 1000.
Default	10
Front Panel	Meas, ACP Meas Setup, Avg Num
<b>45. [:SENSe]:ACPower:AVERAge[:STATe]</b>	
Syntax	[:SENSe]:ACPower:AVERAge[:STATe] OFF ON 0 1 [:SENSe]:ACPower:AVERAge[:STATe]?
Function	Turns on or off the Average function of the adjacent channel power measurement. The query returns "0" or "1".
Default	OFF 0
Front Panel	Meas, ACP Meas Setup, Avg Num, On/Off
<b>46. [:SENSe]:ACPower:AVERAge:TCONtrol</b>	
Syntax	[:SENSe]:ACPower:AVERAge:TCONtrol EXPonential REPeat [:SENSe]:ACPower:AVERAge:TCONtrol?
Function	Sets the average mode of the adjacent channel power measurement. The query returns "EXP" or "REP".
Explanations	<ul style="list-style-type: none"> <li>● In Exp mode, the result is from the exponent average</li> </ul>

	<p>calculation of the current N (specified by [:SENSe]:ACPower:AVERage:COUNT) measurements.</p> <ul style="list-style-type: none"> <li>● In REP mode, the result is the arithmetic average of the current N (specified by [:SENSe]:ACPower:AVERage:COUNT) measurements.</li> </ul>
Default	EXponential
Front Panel	Meas, ACP Meas Setup, Avg Mode, Exp/Repeat
<b>47. [:SENSe]:ACPower:BANDwidth:INTegration</b>	
Syntax	[:SENSe]:ACPower:BANDwidth:INTegration <freq> [:SENSe]:ACPower:BANDwidth:INTegration?
Function	Sets the bandwidth of the main channels, the default unit is Hz, such as :ACP:BAND:INT 1000000 or :ACP:BAND:INT 1MHZ The query returns "1000000".
Explanations	<freq> ranges from 33 Hz to 1 GHz.
Default	2 MHz
Front Panel	Meas, ACP Meas Setup, Main CH BW
<b>48. [:SENSe]:ACPower:BANDwidth:ACHannel</b>	
Syntax	[:SENSe]:ACPower:BANDwidth:ACHannel <freq> [:SENSe]:ACPower:BANDwidth:ACHannel?
Function	Sets the bandwidth of the adjacent channel, the default unit is Hz, such as :ACP:BAND:ACH 1000000 or :ACP:BAND:ACH 1MHZ The query returns "1000000".
Explanations	<freq> ranges from 33 Hz to 1 GHz.
Default	2 MHz
Front Panel	Meas, ACP Meas Setup, Adj CH BW
<b>49. [:SENSe]:ACPower:CSPacing</b>	
Syntax	[:SENSe]:ACPower:CSPacing <freq> [:SENSe]:ACPower:CSPacing?
Function	Sets the center frequency difference between the main and

	adjacent channels, the default unit is Hz, such as :ACP:BAND:CSP 1000000 or :ACP:BAND:CSP 1MHZ The query returns "1000000".
Explanations	<freq> ranges from 33 Hz to 1 GHz.
Default	2 MHz
Front Panel	Meas, ACP Meas Setup, CH Spacing
<b>50. [:SENSe]:CHPower:AVERage:COUNT</b>	
Syntax	[:SENSe]:CHPower:AVERage:COUNT <integer> [:SENSe]:CHPower:AVERage:COUNT?
Function	Sets the number of averages of the channel power measurement results, such as :CHP:AVER:COUN 100 The query returns "100".
Explanations	<integer> ranges from 1 to 1000.
Default	10
Front Panel	Meas, Chan Pwr Meas Setup, Avg Num
<b>51. [:SENSe]:CHPower:AVERage[:STATe]</b>	
Syntax	[:SENSe]:CHPower:AVERage[:STATe] OFF ON 0 1 [:SENSe]:CHPower:AVERage[:STATe]?
Function	Turns on or off the average function of the channel power measurement. The query returns "0" or "1".
Default	OFF 0
Front Panel	Meas, Chan Pwr Meas Setup, Avg Num, On/Off
<b>52. [:SENSe]:CHPower:AVERage:TCONtrol</b>	
Syntax	[:SENSe]:CHPower:AVERage:TCONtrol EXPonential REPeat [:SENSe]:CHPower:AVERage:TCONtrol?
Function	Sets the average mode of the channel power measurement. The query returns "EXP" or "REP".
Explanations	<ul style="list-style-type: none"> <li>● In Exp mode, the result is from the exponent average calculation of the current N (specified by</li> </ul>

	<p>[:SENSe]:CHPower:AVERAge:COUNT measurements.</p> <ul style="list-style-type: none"> <li>In REP mode, the result is the arithmetic average of the current N (specified by [:SENSe]:CHPower:AVERAge:COUNT) measurements.</li> </ul>
Default	EXPonential
Front Panel	Meas, Chan Pwr Meas Setup, Avg Mode, Exp/Repeat
<b>53. [:SENSe]:CHPower:BANDwidth:INTEgration</b>	
Syntax	[:SENSe]:CHPower:BANDwidth:INTEgration <freq> [:SENSe]:CHPower:BANDwidth:INTEgration?
Function	Sets the integration bandwidth for the channel power measurement, the default unit is Hz, such as :CHP:BAND:INT 1000000 or :CHP:BAND:INT 1MHZ The query returns "1000000".
Explanations	<freq> ranges from 100 Hz to 3 GHz.
Default	2 MHz
Front Panel	Meas, Chan Pwr Meas Setup, Integ BW
<b>54. [:SENSe]:CHPower:FREQuency:SPAN</b>	
Syntax	[:SENSe]:CHPower:FREQuency:SPAN <freq> [:SENSe]:CHPower:FREQuency:SPAN?
Function	Sets the channel span for the channel power measurement, the default unit is Hz, such as :CHP:FREQ:SPAN 3000000 or :CHP:FREQ:SPAN 3MHZ The query returns "3000000".
Explanations	<freq> ranges from 100 Hz to 3 GHz.
Default	3 MHz
Front Panel	Meas, Chan Pwr Meas Setup, CH Pwr Span
<b>55. [:SENSe]:CNRatio:AVERAge:COUNT</b>	
Syntax	[:SENSe]:CNRatio:AVERAge:COUNT <integer> [:SENSe]:CNRatio:AVERAge:COUNT?
Function	Sets the number of averages of the C/N Ratio measurement results, such as

	:CNR:AVER:COUN 100 The query returns "100".
Explanations	<integer> ranges from 1 to 1000.
Default	10
Front Panel	Meas, C/N Ratio Meas Setup, Avg Num
<b>56. [:SENSe]:CNRatio:AVERage[:STATe]</b>	
Syntax	[:SENSe]:CNRatio:AVERage[:STATe] OFF ON 0 1 [:SENSe]:CNRatio:AVERage[:STATe]?
Function	Turns on or off the average function of the C/N Ratio measurement. The query returns "0" or "1".
Default	OFF 0
Front Panel	Meas, C/N Ratio Meas Setup, Avg Num, On/Off
<b>57. [:SENSe]:CNRatio:AVERage:TCONtrol</b>	
Syntax	[:SENSe]:CNRatio:AVERage:TCONtrol EXPonential REPeat [:SENSe]:CNRatio:AVERage:TCONtrol?
Function	Sets the average mode of the C/N Ratio measurement. The query returns "EXP" or "REP".
Explanations	<ul style="list-style-type: none"> <li>● In Exp mode, the result is from the exponent average calculation of the current N (specified by [:SENSe]:CNRatio:AVERage:COUNT) measurements.</li> <li>● In REP mode, the result is the arithmetic average of the current N (specified by [:SENSe]:CNRatio:AVERage:COUNT) measurements.</li> </ul>
Default	EXPonential
Front Panel	Meas, C/N Ratio Meas Setup, Avg Mode, Exp/Repeat
<b>58. [:SENSe]:CNRatio:BANDwidth:INTegration</b>	
Syntax	[:SENSe]:CNRatio:BANDwidth:INTegration <freq> [:SENSe]:CNRatio:BANDwidth:INTegration?
Function	Sets the carrier bandwidth, the default unit is Hz, such as :CNR:BAND:INT 1000000 or :CNR:BAND:INT 1MHZ The query returns "1000000".

Explanations	<freq> ranges from 33 Hz to 1 GHz.
Default	2 MHz
Front Panel	Meas, C/N Ratio Meas Setup, Carrier BW
<b>59. [:SENSe]:CNRatio:BANDwidth:NOISe</b>	
Syntax	[:SENSe]:CNRatio:BANDwidth:NOISe <freq> [:SENSe]:CNRatio:BANDwidth:NOISe?
Function	Sets the noise bandwidth, the default unit is Hz, such as :CNR:BAND:NOIS 1000000 or :CNR:BAND:NOIS 1MHZ The query returns "1000000".
Explanations	<freq> ranges from 33 Hz to 1 GHz.
Default	2 MHz
Front Panel	Meas, C/N Ratio Meas Setup, Noise BW
<b>60. [:SENSe]:CNRatio:OFFSet</b>	
Syntax	[:SENSe]:CNRatio:OFFSet <freq> [:SENSe]:CNRatio:OFFSet?
Function	Sets the center frequency offset between the carrier and noise, the default unit is Hz, such as :CNR:OFFS 1000000 or :CNR:OFFS 1MHZ The query returns "1000000".
Explanations	<freq> ranges from 33 Hz to 1 GHz.
Default	2 MHz
Front Panel	Meas, C/N Ratio Meas Setup, Offset Freq
<b>61. [:SENSe]:EBWidth:AVERage:COUNT</b>	
Syntax	[:SENSe]:EBWidth:AVERage:COUNT <integer> [:SENSe]:EBWidth:AVERage:COUNT?
Function	Sets the number of averages of the Emission Bandwidth measurement results, such as :EBW:AVER:COUN 100 The query returns "100".
Explanations	<integer> ranges from 1 to 1000.
Default	10
Front Panel	Meas, EBW

	Meas Setup, Avg Num
<b>62. [:SENSe]:EBWidth:AVERage[:STATe]</b>	
Syntax	[:SENSe]:EBWidth:AVERage[:STATe] OFF ON 0 1 [:SENSe]:EBWidth:AVERage[:STATe]?
Function	Turns on or off the average function of the Emission Bandwidth measurement. The query returns "0" or "1".
Default	OFF 0
Front Panel	Meas, EBW Meas Setup, Avg Num, On/Off
<b>63. [:SENSe]:EBWidth:AVERage:TCONtrol</b>	
Syntax	[:SENSe]:EBWidth:AVERage:TCONtrol EXPonential REPeat [:SENSe]:EBWidth:AVERage:TCONtrol?
Function	Sets the average mode of the Emission Bandwidth measurement. The query returns "EXP" or "REP".
Explanations	<ul style="list-style-type: none"> <li>● In Exp mode, the result is from the exponent average calculation of the current N (specified by [:SENSe]:EBWidth:AVERage:COUNT) measurements.</li> <li>● In REP mode, the result is the arithmetic average of the current N (specified by [:SENSe]:EBWidth:AVERage:COUNT) measurements.</li> </ul>
Default	● EXPonential
Front Panel	Meas, EBW Meas Setup, Avg Mode, Exp/Repeat
<b>64. [:SENSe]:EBWidth:MAXHold:STATe</b>	
Syntax	[:SENSe]:EBWidth:MAXHold:STATe OFF ON 0 1 [:SENSe]:EBWidth:MAXHold:STATe?
Function	Turns on or off the Max Hold function. The query returns "0" or "1".
Default	OFF 0
Front Panel	Meas, EBW Meas Setup, Max Hold, On/Off
<b>65. [:SENSe]:EBWidth:FREQUency:SPAN</b>	

Syntax	[[:SENSe]:EBWidth:FREQuency:SPAN <freq> [:SENSe]:EBWidth:FREQuency:SPAN?
Function	Sets the span of the Emission Bandwidth measurement, the default unit is Hz, such as :EBW:FREQ:SPAN 1000000 or :EBW:FREQ:SPAN 1MHZ The query returns "1000000".
Explanations	<freq> ranges from 100 Hz to 3 GHz.
Default	2 MHz
Front Panel	Meas, EBW Meas Setup, Span
<b>66. [:SENSe]:EBWidth:XDB</b>	
Syntax	[[:SENSe]:EBWidth:XDB <real_amp> [:SENSe]:EBWidth:XDB?
Function	Sets the value of X dB to calculate the emission bandwidth, such as :EBWidth:XDB -20 The query returns "-2.000000E+01".
Explanations	<real_amp> ranges from -100 dB to -0.1 dB.
Default	-10 dB
Front Panel	Meas, EBW Meas Setup, X dB
<b>67. [:SENSe]:HDISt:AVERAge:COUNT</b>	
Syntax	[[:SENSe]:HDISt:AVERAge:COUNT <integer> [:SENSe]:HDISt:AVERAge:COUNT?
Function	Sets the number of averages of the Harmonic Distortion measurement results, such as :HDIS:AVER:COUN 100 The query returns "100".
Explanations	<integer> ranges from 1 to 1000.
Default	10
Front Panel	Meas, Harmo Dist Meas Setup, Avg Num
<b>68. [:SENSe]:HDISt:AVERAge[:STATe]</b>	
Syntax	[[:SENSe]:HDISt:AVERAge[:STATe] OFF ON 0 1 [:SENSe]:HDISt:AVERAge[:STATe]?

Function	Turns on or off the average function of the Harmonic Distortion measurement. The query returns "0" or "1".
Default	OFF 0
Front Panel	Meas, Harmo Dist Meas Setup, Avg Num, On/Off
<b>69. [:SENSe]:HDISt:AVERAge:TCONtrol</b>	
Syntax	[:SENSe]:HDISt:AVERAge:TCONtrol EXPonential REPeat [:SENSe]:HDISt:AVERAge:TCONtrol?
Function	Sets the average mode of the Harmonic Distortion measurement. The query returns "EXP" or "REP".
Explanations	<ul style="list-style-type: none"> <li>● In Exp mode, the result is from the exponent average calculation of the current N (specified by [:SENSe]:HDISt:AVERAge:COUNT) measurements.</li> <li>● In REP mode, the result is the arithmetic average of the current N (specified by [:SENSe]:HDISt:AVERAge:COUNT) measurements.</li> </ul>
Default	● EXPonential
Front Panel	Meas, Harmo Dist Meas Setup, Avg Mode, Exp/Repeat
<b>70. [:SENSe]:HDISt:NUMBers</b>	
Syntax	[:SENSe]:HDISt:NUMBers <integer> [:SENSe]:HDISt:NUMBers?
Function	Sets the number of harmonics for carrier measurement, such as :HDIS:NUMB 5 The query returns "5".
Explanations	<integer> ranges from 2 to 10.
Default	10
Front Panel	Meas, Harmo Dist Meas Setup, NO.of Harmo
<b>71. [:SENSe]:HDISt:TIME</b>	
Syntax	[:SENSe]:HDISt:TIME <time> [:SENSe]:HDISt:TIME?
Function	Sets the sweep time of the Harmonic measurement, the default unit

	is s, such as HDIS:TIME 0.01 The query returns "1.000000E-02".
Explanations	<time> ranges from 20 us to 3 ks.
Default	10 ms
Front Panel	Meas, Harmo Dist Meas Setup, Sweep Time
<b>72. [:SENSe]:HDIS:TIME:AUTO[:STATe]</b>	
Syntax	[:SENSe]:HDIS:TIME:AUTO[:STATe] OFF ON 0 1 [:SENSe]:HDIS:TIME:AUTO[:STATe]?
Function	Turns on or off the Auto function of sweep time for the harmonic measurement. The query returns "0" or "1".
Default	ON 1
Front Panel	Meas, Harmo Dist Meas Setup, Sweep Time, Auto/Manual
<b>73. [:SENSe]:OBWidth:AVERage:COUNT</b>	
Syntax	[:SENSe]:OBWidth:AVERage:COUNT <integer> [:SENSe]:OBWidth:AVERage:COUNT?
Function	Sets the number of averages of the Occupied Bandwidth measurement results, such as :OBW:AVER:COUN 100 The query returns "100".
Explanations	<integer> ranges from 1 to 1000.
Default	10
Front Panel	Meas, OBW Meas Setup, Avg Num
<b>74. [:SENSe]:OBWidth:AVERage [:STATe]</b>	
Syntax	[:SENSe]:OBWidth:AVERage [:STATe] OFF ON 0 1 [:SENSe]:OBWidth:AVERage [:STATe]?
Function	Turns on or off the average function of the Occupied Bandwidth measurement. The query returns "0" or "1".
Default	OFF 0

Front Panel	Meas, OBW Meas Setup, Avg Num, On/Off
<b>75. [:SENSe]:OBWidth:AVERage:TCONtrol</b>	
Syntax	[:SENSe]:OBWidth:AVERage:TCONtrol EXPonential REPeat [:SENSe]:OBWidth:AVERage:TCONtrol?
Function	Sets the average mode of the Occupied Bandwidth measurement. The query returns "EXP" or "REP".
Explanations	<ul style="list-style-type: none"> <li>● In Exp mode, the result is from the exponent average calculation of the current N (specified by [:SENSe]:OBWidth:AVERage:COUNT) measurements.</li> <li>● In REP mode, the result is the arithmetic average of the current N (specified by [:SENSe]:OBWidth:AVERage:COUNT) measurements.</li> </ul>
Default	● EXPonential
Front Panel	Meas, OBW Meas Setup, Avg Mode, Exp/Repeat
<b>76. [:SENSe]:OBWidth:MAXHold:STATe</b>	
Syntax	[:SENSe]:OBWidth:MAXHold:STATe OFF ON 0 1 [:SENSe]:OBWidth:MAXHold:STATe?
Function	Turns on or off the Max Hold function of the Occupied Bandwidth measurement. The query returns "0" or "1".
Default	OFF 0
Front Panel	Meas, OBW Meas Setup, Max Hold, On/Off
<b>77. [:SENSe]:OBWidth:FREQuency:SPAN</b>	
Syntax	[:SENSe]:OBWidth:FREQuency:SPAN <freq> [:SENSe]:OBWidth:FREQuency:SPAN?
Function	Sets the span of the Occupied Bandwidth measurement, the default unit is Hz, such as :OBW:FREQ:SPAN 1000000 or :OBW:FREQ:SPAN 1MHZ The query returns "1000000".
Explanations	<freq> ranges from 100 Hz to 3 GHz.
Default	2MHz

Front Panel	Meas, OBW Meas Setup, Span
<b>78. [:SENSe]:OBWidth:PERCent</b>	
Syntax	[:SENSe]:OBWidth:PERCent <real> [:SENSe]:OBWidth:PERCent?
Function	Sets the percent of the signal power in the whole frequency span power, such as :OBW:PERC 90 The query returns "9.000000E+01".
Explanations	<real> ranges from 1 to 99.99.
Default	99 (99%)
Front Panel	Meas, OBW Meas Setup, Power Ratio
<b>79. [:SENSe]:ROSCillator:SOURce</b>	
Syntax	[:SENSe]:ROSCillator:SOURce INTernal EXTernal [:SENSe]:ROSCillator:SOURce?
Function	Selects the internal or external reference source. The query returns "INT" or "EXT".
Default	INTernal
Front Panel	System, Ref Source, Int/Ext
<b>80. [:SENSe]:STATus:ANALyzer?</b>	
Syntax	[:SENSe]:STATus:ANALyzer?
Function	Quires the measurement settings of the analyzer.
Front Panel	System, Status, Analyzer
<b>81. [:SENSe]:STATus:CORRections?</b>	
Syntax	[:SENSe]:STATus:CORRections?
Function	Quires the correction status.
Front Panel	System, Status, Corrections
<b>82. [:SENSe]:STATus:DEMod?</b>	
Syntax	[:SENSe]:STATus:DEMod?
Function	Quires the setting status of the demodulation.
Front Panel	System, Status, Demod

<b>83. [:SENSe]:STATus:POWer?</b>	
Syntax	[:SENSe]:STATus:POWer?
Function	Quires the setting status of the power measurement.
Front Panel	System, Status, Power
<b>84. [:SENSe]:STATus:TG?</b>	
Syntax	[:SENSe]:STATus:TG?
Function	Quires the setting status of TG (Tracking Generator).
Front Panel	System, Status, TG
<b>85. [:SENSe]:TOI:AVERAge:COUNT</b>	
Syntax	[:SENSe]:TOI:AVERAge:COUNT <integer> [:SENSe]:TOI:AVERAge:COUNT?
Function	Sets the number of averages of the TOI measurement results, such as :TOI:AVER:COUN 100 The query returns "100".
Explanations	<integer> ranges from 1 to 1000.
Default	10
Front Panel	Meas, TOI Meas Setup, Avg Num
<b>86. [:SENSe]:TOI:AVERAge[:STATe]</b>	
Syntax	[:SENSe]:TOI:AVERAge[:STATe] OFF ON 0 1 [:SENSe]:TOI:AVERAge[:STATe]?
Function	Turns on or off the average function of the TOI measurement. The query returns "0" or "1".
Default	OFF 0
Front Panel	Meas, TOI Meas Setup, Avg Num, On/Off
<b>87. [:SENSe]:TOI:AVERAge:TCONtrol</b>	
Syntax	[:SENSe]:TOI:AVERAge:TCONtrol EXPonential REPeat [:SENSe]:TOI:AVERAge:TCONtrol?
Function	Selects an average mode of the TOI measurement from EXP or REP. The query returns "EXP" or "REP".

Explanations	<ul style="list-style-type: none"> <li>● In Exp mode, the result is from the exponent average calculation of the current N (specified by [:SENSe]:TOI:AVERAge:COUNT) measurements.</li> <li>● In REP mode, the result is the arithmetic average of the current N (specified by [:SENSe]:TOI:AVERAge:COUNT) measurements.</li> </ul>
Default	● EXPonential
Front Panel	Meas, TOI Meas Setup, Avg Num, Exp/Repeat

**88. [:SENSe]:TOI:FREQuency:SPAN**

Syntax	[:SENSe]:TOI:FREQuency:SPAN <freq> [:SENSe]:TOI:FREQuency:SPAN?
Function	Sets the span of the TOI measurement, the default unit is Hz, such as :TOI:FREQ:SPAN 1000000 or :TOI:FREQ:SPAN 1MHZ The query returns 1000000.
Explanations	<freq> ranges from 100 Hz to 3 GHz.
Default	2 MHz
Front Panel	Meas, TOI Meas Setup, Span

**89. [:SENSe]:TPOWer:AVERAge:COUNT**

Syntax	[:SENSe]:TPOWer:AVERAge:COUNT <integer> [:SENSe]:TPOWer:AVERAge:COUNT?
Function	Sets the number of averages of the T-Power measurement results, such as :TPOW:AVER:COUN 100 The query returns "100".
Explanations	<integer> ranges from 1 to 1000.
Default	10
Front Panel	Meas, T-Power Meas Setup, Avg Num

**90. [:SENSe]:TPOWer:AVERAge[:STATe]**

Syntax	[:SENSe]:TPOWer:AVERAge[:STATe] OFF ON 0 1 [:SENSe]:TPOWer:AVERAge[:STATe]?
--------	--

Function	Turns on or off the average function of the Time-domain Power measurement. The query returns "0" or "1".
Default	OFF 0
Front Panel	Meas, T-Power Meas Setup, Avg Num, On/Off
<b>91. [:SENSe]:TPOWer:AVERAge:TCONtrol</b>	
Syntax	[:SENSe]:TPOWer:AVERAge:TCONtrol EXPonential REPeat [:SENSe]:TPOWer:AVERAge:TCONtrol?
Function	Sets the average mode of the T-Power measurement.
Explanations	<ul style="list-style-type: none"> <li>● In Exp mode, the result is from the exponent average calculation of the current N (specified by [:SENSe]:TPOWer:AVERAge:COUNT) measurements.</li> <li>● In REP mode, the result is the arithmetic average of the current N (specified by [:SENSe]:TPOWer:AVERAge:COUNT) measurements.</li> </ul>
Default	● EXPonential
Front Panel	Meas, T-Power Meas Setup, Avg Mode, Exp/Repeat
<b>92. [:SENSe]:TPOWer:MODE</b>	
Syntax	[:SENSe]:TPOWer:MODE AVERAge PEAK RMS [:SENSe]:TPOWer:MODE?
Function	Sets the type of the Time-domain Power measurement. The query returns "AVER", "PEAK" or "RMS".
Default	PEAK
Front Panel	Meas, T-Power Meas Setup, Peak/Average/RMS
<b>93. [:SENSe]:TPOWer:LLIMit</b>	
Syntax	[:SENSe]:TPOWer:LLIMit <time> [:SENSe]:TPOWer:LLIMit?
Function	Sets the left limit of the Time-domain Power measurement, the default unit is s, such as :TPOW:LLIM 0.005 The query returns "5.000000E-03".

Explanations	<time> ranges from 0 to Right limit.
Default	0 s
Front Panel	Meas, T-Power Meas Setup, Start Line
<b>94. [:SENSe]:TPOWer:RLIMit</b>	
Syntax	[:SENSe]:TPOWer:RLIMit <time> [:SENSe]:TPOWer:RLIMit?
Function	Sets the right limit of the Time-domain Power measurement, the default unit is s, such as :TPOW:RLIM 0.01 The query returns "1.000000E-02".
Explanations	<time> ranges from the left limit to Sweep Time.
Default	100 ms
Front Panel	Meas, T-Power Meas Setup, Stop Line

## :SOURce

- :SOURce:CORRection:OFFSet
- :SOURce:FREQuency:OFFSet
- :SOURce:POWer:LEVel:IMMediate:AMPLitude
- :SOURce:POWer:MODE
- :SOURce:POWer:SPAN
- :SOURce:POWer:START
- :SOURce:POWer:SWEEp
- :SOURce:TRACe:STORref
- :SOURce:TRACe:REF:STATe

**NOTE: The SOURce commands only apply to DSA1030 which has been installed the tracking generator option.**

<b>1. :SOURce:CORRection:OFFSet</b>	
Syntax	:SOURce:CORRection:OFFSet <rel_ampl> :SOURce:CORRection:OFFSet?
Function	Sets the output amplitude offset for the tracking source in dB, such as :SOUR:CORR:OFFS 10 The query returns "1.000000E+01".
Explanations	<rel_ampl> ranges from -200 dB to 200 dB.
Default	0 dB
Front Panel	Source, TG Lvl Offset
<b>2. :SOURce:FREQuency:OFFSet</b>	
Syntax	:SOURce:FREQuency:OFFSet <freq> :SOURce:FREQuency:OFFSet?
Function	Sets the output frequency offset for the tracking source, the default unit is Hz, such as :SOUR:FREQ:OFFS 1000000 The query returns "1000000".
Explanations	<freq> ranges from -300 MHz to 300 MHz.
Default	0Hz
Front Panel	Source, Freq Offset
<b>3. :SOURce:POWer:LEVel:IMMEdiate:AMPLitude</b>	
Syntax	:SOURce:POWer:LEVel:IMMEdiate:AMPLitude <ampl> :SOURce:POWer:LEVel:IMMEdiate:AMPLitude?
Function	Sets the output power of the tracking source when the "Power Sweep" is off, the default unit is dBm, such as :SOUR:POW:LEV:IMM:AMP -10 The query returns "-1.000000E+01".
Explanations	<ampl> ranges from -20 dBm to 0 dBm when the parameter <rel_ampl> in the :SOURce:CORRection:OFFSet command is 0, or else the range should plus the value of <rel_ampl>.
Default	0 dBm
Front Panel	Source, Power Sweep, Off, TG Level
<b>4. :SOURce:POWer:MODE</b>	
Syntax	:SOURce:POWer:MODE FIXEd SWEep

	:SOURce:POWer:MODE?
Function	Sets the power output mode. The query returns "FIX" or "SWE".
Default	FIXed
Front Panel	Power sweep output: Source, Power Sweep, On Fixed sweep output: Source, Power Sweep, Off
<b>5. :SOURce:POWer:SPAN</b>	
Syntax	:SOURce:POWer:SPAN <rel_ampl> :SOURce:POWer:SPAN?
Function	Sets the output amplitude range of the tracking source in dB when the "Power Sweep" is off , such as :SOUR:POW:SPAN 10 The query returns "1.000000E+01".
Explanations	<rel_ampl> ranges from 0 dB to 20 dB.
Default	0 dB
Front Panel	Source, Power Sweep, Off, Power Range
<b>6. :SOURce:POWer:STARt</b>	
Syntax	:SOURce:POWer:STARt <ample> :SOURce:POWer:STARt?
Function	Sets the output power of the tracking source when the "Power Sweep" is on, the default unit is dBm, such as :SOUR:POW:STAR -10 The query returns "-1.000000E+01".
Explanations	<ample> ranges from -20 dBm to 0 dBm when the parameter <rel_ampl> in the :SOURce:CORRection:OFFSet command is 0, or else the range should plus the value of <rel_ampl>.
Default	0 dBm
Front Panel	Source, Power Sweep, On, TG Level
<b>7. :SOURce:POWer:SWEep</b>	
Syntax	:SOURce:POWer:SWEep <rel_ampl> :SOURce:POWer:SWEep?
Function	Sets the output power range of the tracking source in dB when the "Power Sweep" is on, such as :SOUR:POW:SWE 10

	The query returns "1.000000E+01".
Explanations	<rel_ampl> ranges from 0 dB to 20 dB.
Default	0 dB
Front Panel	Source, Power Sweep, On, Power Range
<b>8. :SOURce:TRACe:STORref</b>	
Syntax	:SOURce:TRACe:STORref
Function	Save the normalized trace.
Front Panel	Source, Normalize, Stor Ref
<b>9. :SOURce:TRACe:REF:STATe</b>	
Syntax	:SOURce:TRACe:REF:STATe OFF ON 0 1 :SOURce:TRACe:REF:STATe?
Function	Sets if to display the reference trace. The query returns "0" or "1".
Default	OFF 0
Front Panel	Source, Normalize, Ref Trace, View/Blank

## **:STATus**

- :STATus:OPERation:CONDition?
- :STATus:OPERation:ENABle
- :STATus:OPERation[:EVENT]?
- :STATus:PRESet
- :STATus:QUEStionable:CONDition?
- :STATus:QUEStionable:ENABle
- :STATus:QUEStionable[:EVENT]?

<b>1. :STATus:OPERation:CONDition?</b>	
Syntax	:STATus:OPERation:CONDition?
Function	The query returns the condition register status in the operation status register, such as "24".
Explanations	Bit 15 and bits 12-8 are not used in the Standard Operation Register and always regarded as "0", thereinto this command returns a decimal number that relates to any binary number whose bits of 15 and 12-8 are 0 within 0000000000000000 (0) and 1111111111111111 (32767).
<b>2. :STATus:OPERation:ENABLE</b>	
Syntax	:STATus:OPERation:ENABLE <integer> :STATus:OPERation:ENABLE?
Function	Sets the enable register value in the operation status register, such as: :STAT:OPER:ENAB 100 The query returns "100".
Explanations	Bit 15 and bits 12-8 are not used in the Standard Operation Register and always regarded as "0", thereinto <integer> is a decimal number that relates to any binary number whose bits of 15 and 12-8 are 0 within 0000000000000000 (0) and 1111111111111111 (32767).
Default	0
<b>3. :STATus:OPERation[:EVENT]?</b>	
Syntax	:STATus:OPERation[:EVENT]?
Function	The query returns the event register value in the operation status register, such as "24".
Explanations	Bit 15 and bits 12-8 are not used in the Standard Operation Register and always regarded as "0", thereinto this command returns a decimal number that relates to any binary number whose bits of 15 and 12-8 are 0 within 0000000000000000 (0) and 1111111111111111 (32767).
<b>4. :STATus:PRESet</b>	
Syntax	:STATus:PRESet
Function	Clears the enable register value in both the operation status register

	and questionable status register.
<b>5. :STATus:QUESTIONable:CONDition?</b>	
Syntax	:STATus:QUESTIONable:CONDition?
Function	The query returns the condition register value in the questionable status register, such as "0".
Explanations	Bit 15 and bits 12-9 are not used in the Questionable Data Register and always regarded as "0", thereinto this command returns a decimal number that relates to any binary number whose bits of 15 and 12-9 are 0 within 0000000000000000 (0) and 1111111111111111 (32767).
<b>6. :STATus:QUESTIONable:ENABLE</b>	
Syntax	:STATus:QUESTIONable:ENABLE <integer> :STATus:QUESTIONable:ENABLE?
Function	Sets the enable register value in the questionable status register, such as :STAT:QUES:ENAB 100 The query returns "100".
Explanations	Bit 15 and bits 12-9 are not used in the Questionable Data Register and always regarded as "0", thereinto <integer> is a decimal number that relates to any binary number whose bits of 15 and 12-9 are 0 within 0000000000000000 (0) and 1111111111111111 (32767).
Default	0
<b>7. :STATus:QUESTIONable[:EVENT]?</b>	
Syntax	:STATus:QUESTIONable[:EVENT]?
Function	The query returns the event register value in the questionable status register, such as "0".
Explanations	Bit 15 and bits 12-9 are not used in the Questionable Data Register and always regarded as "0", thereinto this command returns a decimal number that relates to any binary number whose bits of 15 and 12-9 are 0 within 0000000000000000 (0) and 1111111111111111 (32767).

## :SYSTem

- :SYSTem:COMMunicate:APORt
- :SYSTem:COMMunicate:GPIB[:SELF]:ADDRess
- :SYSTem:COMMunicate:LAN[:SELF]:RESet
- :SYSTem:COMMunicate:LAN[:SELF]:DHCP:STATe
- :SYSTem:COMMunicate:LAN[:SELF]:AUToip:STATe
- :SYSTem:COMMunicate:LAN[:SELF]:MANuip:STATe
- :SYSTem:COMMunicate:LAN[:SELF]:IP:ADDRess
- :SYSTem:COMMunicate:LAN[:SELF]:IP:SUBMask
- :SYSTem:COMMunicate:LAN[:SELF]:IP:GATeway
- :SYSTem:COMMunicate:LAN[:SELF]:IP:DNSServer
- :SYSTem:COMMunicate:USB[:SELF]:ADDRess?
- :SYSTem:COMMunicate:USB[:SELF]:CLASs
- :SYSTem:CONFigure:INFomation?
- :SYSTem:CONFigure:MESSage?
- :SYSTem:DATE
- :SYSTem:ERRor[:NEXT]?
- :SYSTem:FSWItch[:STATe]
- :SYSTem:LANGuage
- :SYSTem:LKEY\*\*\*\*\*
- :SYSTem:OPTions?\*\*\*\*\*
- :SYSTem:PON:TYPE
- :SYSTem:PRESet
- :SYSTem:PRESet:TYPE
- :SYSTem:PRESet[:USER]:SAVE
- :SYSTem:PROBe[:STATe]
- :SYSTem:SPEaker[:STATe]
- :SYSTem:SPEaker:VOLume
- :SYSTem:TIME
- :SYSTem:VERSion?

**NOTE: The commands marked with "\*\*\*\*\*" only apply to DSA1030 which has been installed at least one option.**

<b>1. :SYSTem:COMMunicate:APORt</b>	
Syntax	:SYSTem:COMMunicate:APORt GPIB LAN USB OFF :SYSTem:COMMunicate:APORt?
Function	Selects a communication port. The query returns "GPIB", "LAN", "USB" or "OFF".
Default	OFF
Front Panel	System, I/O Setting, Remote I/O, Off/ LAN/USB/GPIB
<b>2. :SYSTem:COMMunicate:GPIB[:SELF]:ADDRess</b>	
Syntax	:SYSTem:COMMunicate:GPIB[:SELF]:ADDRess <integer> :SYSTem:COMMunicate:GPIB[:SELF]:ADDRess?
Function	Sets the GPIB address, such as :SYST:COMM:GPIB:ADDR 16 The query returns "16".
Explanations	<integer> is an integer within 0 and 30.
Default	18
Front Panel	System, I/O Setting, GPIB
<b>3. :SYSTem:COMMunicate:LAN[:SELF]:RESet</b>	
Syntax	:SYSTem:COMMunicate:LAN[:SELF]:RESet
Function	Resets the LXI and all parameters about the LAN.
Front Panel	System, I/O Setting, LAN, Reset
<b>4. :SYSTem:COMMunicate:LAN[:SELF]:DHCP:STATe</b>	
Syntax	:SYSTem:COMMunicate:LAN[:SELF]:DHCP:STATe OFF ON 0 1 :SYSTem:COMMunicate:LAN[:SELF]:DHCP:STATe?
Function	Turns on or off the DHCP. The query returns "0" or "1".
Default	ON 1
Front Panel	System, I/O Setting, LAN, DHCP, On/Off
<b>5. :SYSTem:COMMunicate:LAN[:SELF]:AUTOip:STATe</b>	
Syntax	:SYSTem:COMMunicate:LAN[:SELF]:AUTOip:STATe OFF ON 0 1 :SYSTem:COMMunicate:LAN[:SELF]:AUTOip:STATe?
Function	Turns on or off the Auto IP. The query returns "0" or "1".
Default	ON 1

Front Panel	System, I/O Setting, LAN, Auto-IP, On/Off
<b>6. :SYSTem:COMMunicate:LAN[:SELF]:MANuip:STATe</b>	
Syntax	:SYSTem:COMMunicate:LAN[:SELF]:MANuip:STATe OFF ON 0 1 :SYSTem:COMMunicate:LAN[:SELF]:MANuip:STATe?
Function	Turns on or off the Manual IP. The query returns "0" or "1".
Default	OFF 0
Front Panel	System, I/O Setting, LAN, Manual-IP, On/Off
<b>7. :SYSTem:COMMunicate:LAN[:SELF]:IP:ADDRESS</b>	
Syntax	:SYSTem:COMMunicate:LAN[:SELF]:IP:ADDRESS <ip_address> :SYSTem:COMMunicate:LAN[:SELF]:IP:ADDRESS?
Function	Sets the IP address, such as :SYST:COMM:LAN:IP:ADD 172.16.3.199 The query returns "172.16.3.199".
Explanations	The format of <ip_address > is: nnn.nnn.nnn.nnn, thereinto, the first "nnn" ranges from 0 to 223 (except 127) and the others range from 0 to 255.
Front Panel	System, I/O Setting, LAN, Manual-IP(On), IP, IP Address
<b>8. :SYSTem:COMMunicate:LAN[:SELF]:IP:SUBMask</b>	
Syntax	:SYSTem:COMMunicate:LAN[:SELF]:IP:SUBMask <ip_address> :SYSTem:COMMunicate:LAN[:SELF]:IP:SUBMask?
Function	Sets the submask, such as :SYST:COMM:LAN:IP:SUBM 255.255.255.0 The query returns "255.255.255.0".
Explanations	The format of < ip_address> is: nnn.nnn.nnn.nnn, thereinto, "nnn" ranges from 0 to 255.
Front Panel	System, I/O Setting, LAN, Manual-IP (On), IP, Subnet Mask
<b>9. :SYSTem:COMMunicate:LAN[:SELF]:IP:GATeway</b>	
Syntax	:SYSTem:COMMunicate:LAN[:SELF]:IP:GATeway <ip_address> :SYSTem:COMMunicate:LAN[:SELF]:IP:GATeway?
Function	Specifies a default gateway, such as :SYST:COMM:LAN:IP:GAT 172.16.3.1 The query returns "172.16.3.1".

Explanations	The format of <ip_address >is: nnn.nnn.nnn.nnn, thereinto, the first "nnn" ranges from 0 to 223 (except 127) and the others range from 0 to 255.
Front Panel	System, I/O Setting, LAN, Manual-IP(On), IP, Gateway
<b>10. :SYSTem:COMMunicate:LAN[:SELF]:IP:DNSServer</b>	
Syntax	:SYSTem:COMMunicate:LAN[:SELF]:IP:DNSServer <ip_address> :SYSTem:COMMunicate:LAN[:SELF]:IP:DNSServer?
Function	Sets the DNS server address, such as :SYST:COMM:LAN:IP:DNSS 172.16.2.2 The query returns "172.16.2.2".
Explanations	The format of <ip_address >is: nnn.nnn.nnn.nnn, thereinto, the first "nnn" ranges from 0 to 223 (except 127) and the others range from 0 to 255.
Front Panel	System, I/O Setting, LAN, DNS
<b>11. :SYSTem:COMMunicate:USB[:SELF]:ADDRess?</b>	
Syntax	:SYSTem:COMMunicate:USB[:SELF]:ADDRess?
Function	Views the USB device address.
Default	1
<b>12. :SYSTem:COMMunicate:USB[:SELF]:CLASs</b>	
Syntax	:SYSTem:COMMunicate:USB[:SELF]:CLASs TMC PRINter AUTO :SYSTem:COMMunicate:USB[:SELF]:CLASs?
Function	Sets the USB device type. The query returns "TMC", "PRIN" or "AUTO".
Default	TMC
Front Panel	System, I/O Setting, USB, Dev Class, AutoConfig/TMC/Printer
<b>13. :SYSTem:CONFigure:INFomation?</b>	
Syntax	:SYSTem:CONFigure:INFomation?
Function	The query returns the same system information as those displayed on the screen of the front panel.
Explanations	The system information involves the software and hardware version and so on.
Front Panel	System, Information, System Info

<b>14. :SYSTem:CONFigure:MESSage?</b>	
Syntax	:SYSTem:CONFigure:MESSage?
Function	The query returns the system messages that have appeared recently, such as 10 U Disk connected. 2009-12-15 16:08:05
Explanations	The analyzer can display up to 71 items of history information.
Front Panel	System, Information, System Msg
<b>15. :SYSTem:DATE</b>	
Syntax	:SYSTem:DATE <year>,<month>,<day> :SYSTem:DATE?
Function	Sets the real-time date of the analyzer, such as :SYSTem:DATE 2009,12,29 The query returns "2009,12,29".
Explanations	<ul style="list-style-type: none"> <li>● &lt;year&gt; is a four-digit figure.</li> <li>● &lt;month&gt; is an integer within 01 and 12.</li> <li>● &lt;day&gt; is an integer within 01 and 31.</li> </ul>
Front Panel	System, Time/Date, Set Date
<b>16. :SYSTem:ERRor[:NEXT]?</b>	
Syntax	:SYSTem:ERRor[:NEXT]?
Function	The query returns the error item that first entered the error queue and deletes it, such as -410,"Query INTERRUPTED"
Explanations	You can use *CLS command to clear all information in the error queues. If no error exists, the query returns 0, "No Error"*.
*Remark: The double quotation marks here would be returned.	
<b>17. :SYSTem:FSWItch[:STATe]</b>	
Syntax	:SYSTem:FSWItch[:STATe] OFF ON 0 1 :SYSTem:FSWItch[:STATe]?
Function	Sets if to use the front switch or not. The query returns "0" or "1".
Default	ON 1
Front Panel	System, Front Switch, On/Off
<b>18. :SYSTem:LANGUage</b>	

Syntax	:SYSTem:LANGUage ENGLISH CHINEse :SYSTem:LANGUage?
Function	Sets the display language. The query returns "ENGL" or "CHIN".
Default	ENGLISH
Front Panel	System, Language, English/Chinese
<b>19. :SYSTem:LKEY</b>	
Syntax	:SYSTem:LKEY < "option">, < "license key"> :SYSTem:LKEY? < "option">
Function	Installs and starts the options of the analyzer, such as :SYST:LKEY 0001,6L380045P32POOPK The query returns "001,6L380045P32POOPK".
Explanations	<ul style="list-style-type: none"> <li>• &lt;"option"&gt; is a four-digit figure, such as 0001.</li> <li>• &lt;"license key"&gt; is the option license key. Note each option has a unique license key and can only be used in one analyzer.</li> </ul>
Front Panel	System, License, Active
<b>20. :SYSTem:OPTions?</b>	
Syntax	:SYSTem:OPTions?
Function	The query returns the installed option list, including the option number, the license key and the state of activation.
<b>21. :SYSTem:PON:TYPE</b>	
Syntax	:SYSTem:PON:TYPE PRESet LAST :SYSTem:PON:TYPE?
Function	Specifies a Power-on settings upon rebooting. The query returns "PRES" or "LAST".
Explanations	The presettings contain factory settings and user settings, you can use ":SYSTem:PRESet:TYPE" to select a desired type.
Default	PRESet
Front Panel	System, Reset, Power On, Last/Preset
<b>22. :SYSTem:PRESet</b>	
Syntax	:SYSTem:PRESet
Function	Recalls the system presettings.
Front Panel	Preset

<b>23. :SYSTEM:PRESet:TYPE</b>	
Syntax	:SYSTEM:PRESet:TYPE FACTory USER :SYSTEM:PRESet:TYPE?
Function	Specifies the system preset type. The query returns "FACT" or "USER".
Default	Factory
Front Panel	System, Reset, Preset Type, Factory/User
<b>24. :SYSTEM:PRESet[:USER]:SAVE</b>	
Syntax	:SYSTEM:PRESet[:USER]:SAVE
Function	Saves the user settings.
Front Panel	System, Reset, Preset Type, User, Save Preset
<b>25. :SYSTEM:PROBe[:STATe]</b>	
Syntax	:SYSTEM:PROBe[:STATe] OFF ON 0 1 :SYSTEM:PROBe[:STATe]?
Function	Sets the probe status. The query returns "0" or "1".
Default	OFF 0
<b>26. :SYSTEM:SPEaker[:STATe]</b>	
Syntax	:SYSTEM:SPEaker[:STATe] OFF ON 0 1 :SYSTEM:SPEaker[:STATe]?
Function	Turns on or off the internal speaker. The query returns "0" or "1".
Default	OFF 0
Front Panel	Demod, Demod, AM/FM Demod Setup, Speaker, On/Off
<b>27. :SYSTEM:SPEaker:VOLume</b>	
Syntax	:SYSTEM:SPEaker:VOLume <integer> :SYSTEM:SPEaker:VOLume?
Function	Adjusts the volume of the speaker.
Explanations	<integer> ranges from 0 to 255.
Default	100
Front Panel	Demod, Demod, AM/FM

	Demod Setup, Speaker, On/Off
<b>28. :SYSTem:TIME</b>	
Syntax	:SYSTem:TIME <hour>,<minute>,<second> :SYSTem:TIME?
Function	Sets the real time of the analyzer clock, such as :SYSTem:TIME 15,50,00 The query returns "15,50,00".
Explanations	<ul style="list-style-type: none"> <li>● &lt;hour&gt; is an integer within 00 and 23.</li> <li>● &lt;minute&gt; is an integer within 00 and 59.</li> <li>● &lt;second&gt; is an integer within 00 and 59.</li> </ul>
Front Panel	System, Time/Date, Set Time
<b>29. :SYSTem:VERSion?</b>	
Syntax	:SYSTem:VERSion?
Function	The query returns the SCPI version number of the analyzer: 1999.0.

## **:TRACe**

- :TRACe[:DATA]
- :TRACe:MATH:PEAK[:DATA]?
- :TRACe:MATH:PEAK:POINts?
- :TRACe:MATH:PEAK:SORT
- :TRACe:MATH:PEAK:TABLE:STATe
- :TRACe:MATH:PEAK:THReshold
- :TRACe:MATH:STATe
- :TRACe:MATH:A
- :TRACe:MATH:B
- :TRACe:MATH:CONST
- :TRACe:MATH:TYPE
- :TRACe<n>:MODE
- :TRACe:AVERAge:CLEAr
- :TRACe:AVERAge:COUNt
- :TRACe:AVERAge:COUNt:CURRent?
- :TRACe<n>:AVERAge:TYPE
- :TRACe:CLEAr:ALL

<b>1. :TRACe[:DATA]</b>	
Syntax	:TRACe[:DATA] TRACE1 TRACE2 TRACE3 TRACE4,<definite_length_block>  <comma_separated_ASCII_data> :TRACe[:DATA]? TRACE1 TRACE2 TRACE3 TRACE4
Function	Loads the user data into the specified tarce. The query returns the specified trace data.
Explanations	<ul style="list-style-type: none"> <li>● You can set the user data by sending ":FORMat[:TRACe][:DATA]".</li> <li>● &lt;definite_length_block&gt;: first executes the command ":FORM REAL".</li> <li>● &lt;comma_separated_ASCII_data&gt;: first executes the command ":FORM ASCII".</li> <li>● The query of ":TRACe? TRACE2" and ":TRACe? TRACE3" will return the upper limit line data and lower limit line data respectively when the Pass/Fail function is On.</li> </ul>
<b>2. :TRACe:MATH:PEAK[:DATA]?</b>	
Syntax	:TRACe:MATH:PEAK[:DATA]?
Function	Queries the frequency (Hz) and amplitude (has the same unit as Y-axis's) from Peak Table. The query returns 10 paris of frequency and amplitude values at most, such as 43500000,-7.253288E+01,43950000,-7.169086E+01
Explanations	<ul style="list-style-type: none"> <li>● The command only queries trace 1.</li> <li>● You can sort the peaks by using ":TRACe:MATH:PEAK:SORT".</li> </ul>
<b>3. :TRACe:MATH:PEAK:POINTs?</b>	
Syntax	:TRACe:MATH:PEAK:POINTs?
Function	Queries the peak readout points from the Peak Table and returns an integer within 0 and 10.
<b>4. :TRACe:MATH:PEAK:SORT</b>	
Syntax	:TRACe:MATH:PEAK:SORT AMPLitude FREQuency :TRACe:MATH:PEAK:SORT?
Function	Sorts the Peak Table. The query returns "AMPL" or "FREQ".
Default	FREQuency

Front Panel	Peak, Peak Table, Peak Sort, Freq/Ampl
<b>5. :TRACe:MATH:PEAK:TABLE:STATe</b>	
Syntax	:TRACe:MATH:PEAK:TABLE:STATe OFF ON 0 1 :TRACe:MATH:PEAK:TABLE:STATe?
Function	Turns on or off the Peak Table. The query returns "0" or "1".
Default	OFF 0
Front Panel	Peak, Peak Table, State, On/Off
<b>6. :TRACe:MATH:PEAK:THReshold</b>	
Syntax	:TRACe:MATH:PEAK:THReshold NORMal DLMore DLLess :TRACe:MATH:PEAK:THReshold?
Function	Sets the readout type of the peak from Peak Table. The query returns "NORM", "DLM" or "DLL".
Explanations	<ul style="list-style-type: none"> <li>● NORMal: Displaying the first ten qualified peaks in the table.</li> <li>● DLMore: Displaying the first ten peaks that not only meet the parameter conditions but also have an amplitude greater than the specified Display Line in the table.</li> <li>● DLLess: Displaying the first ten peaks that not only meet the parameter conditions but also have an amplitude smaller than the specified Display Line in the table.</li> </ul>
Default	NORMal
Front Panel	Peak, Peak Table, Pk Readout, Normal/>DL/<DL
<b>7. :TRACe:MATH:STATe</b>	
Syntax	:TRACe:MATH:STATe OFF ON 0 1 :TRACe:MATH:STATe?
Function	Turns on or off the math trace. The query returns "0" or "1".
Default	OFF 0
Front Panel	Trace, Trace Math, Operate, On/Off
<b>8. :TRACe:MATH:A</b>	
Syntax	:TRACe:MATH:A T1 T2 T3 :TRACe:MATH:A?
Function	Selects a trace as the trace A for the trace math from T1, T2 or T3.

	The query returns "T1", "T2" or "T3".
Default	T1
<b>9. :TRACe:MATH:B</b>	
Syntax	:TRACe:MATH:B T1 T2 T3 :TRACe:MATH:B?
Function	Selects a trace as the trace B for the trace math from T1, T2 or T3. The query returns "T1", "T2" or "T3".
Default	T2
<b>10. :TRACe:MATH:CONSt</b>	
Syntax	:TRACe:MATH:CONSt <integer> :TRACe:MATH:CONSt?
Function	Sets the constant of the trace math in dB, such as :TRAC:MATH:CONS 10 The query returns "1.000000E+01".
Explanations	<integer> ranges from -300 dB to 300 dB.
Default	0 dB
<b>11. :TRACe:MATH:TYPE</b>	
Syntax	:TRACe:MATH:TYPE A-B A+CONST A-CONST :TRACe:MATH:TYPE?
Function	Sets the computational method of the math trace. The query returns "A-B", "A+CONST" or "A-CONST".
Default	A-B
<b>12. :TRACe&lt;n&gt;:MODE</b>	
Syntax	:TRACe<n>:MODE WRITe MAXHold MINHold VIEW BLANK VIDeoavg POWeravg :TRACe<n>:MODE?
Function	Sets the type of specified trace. The query returns "WRIT", "MAXH", "MINH", "VIEW", "BLANK", "VID" or "POW".
Explanations	<ul style="list-style-type: none"> <li>• &lt;n&gt; is 1, 2 or 3.</li> <li>• All trace data will be erased after you change the sweep points (by using "[[:SENSe]:SWEep:POINts]").</li> <li>• The query of ":TRACe2:MODE?" and ":TRACe3:MODE?" will</li> </ul>

	return "UPPERL" and "LOWERL" respectively when the Pass/Fail function is On.
Default	WRITE
Front Panel	Trace, Trace Type, Clear Write/Max Hold/Min Hold/Video Avg/Power Avg/Freeze
<b>13. :TRACe:AVERAge:CLEAr</b>	
Syntax	:TRACe:AVERAge:CLEAr
Function	Clears the number of averages.
<b>14. :TRACe:AVERAge:COUNT</b>	
Syntax	:TRACe:AVERAge:COUNT <integer> :TRACe:AVERAge:COUNT?
Function	Sets the number of averages of the trace, such as :TRACe:AVERAge:COUNT 10 The query returns "10".
Explanations	<integer> ranges from 1 to 1000.
Default	100
Front Panel	Trace, Avg Num
<b>15. :TRACe:AVERAge:COUNT:CURRENT?</b>	
Syntax	:TRACe:AVERAge:COUNT:CURRENT?
Function	The query returns the current number of averages of the trace, such as "1".
<b>16. :TRACe&lt;n&gt;:AVERAge:TYPE</b>	
Syntax	:TRACe<n>:AVERAge:TYPE VIDEO RMS :TRACe<n>:AVERAge:TYPE?
Function	Sets the average type of the trace. The query returns "VID" or "RMS".
Explanations	<ul style="list-style-type: none"> <li>• &lt;n&gt; is 1, 2 or 3.</li> <li>• The query of ":TRACe2:AVERAge:TYPE?" and ":TRACe3:AVERAge:TYPE?" will return "ERR" when the Pass/Fail function is On.</li> </ul>
Default	VIDEO
Front Panel	Trace, Trace Type, Video Avg/Power Avg

<b>17. :TRACe:CLEAr:ALL</b>	
Syntax	:TRACe:CLEAr:ALL
Function	Clears all traces, that is, set the type of all traces to BLANK (Off).
Front Panel	Trace, Clear All

## :TRIGger

- :TRIGger:SEQuence:EXTErnal:SLOPe
- :TRIGger:SEQuence:SOURce
- :TRIGger:SEQuence:VIDeo:LEVel

<b>1. :TRIGger:SEQuence:EXTErnal:SLOPe</b>	
Syntax	:TRIGger:SEQuence:EXTErnal:SLOPe Positive NEGative :TRIGger:SEQuence:EXTErnal:SLOPe?
Function	Sets the external trigger edge. The query returns "POS" or "NEG".
Default	POSitive
Front Panel	Trig, Trig Type, External Trig Setup, Edge, Positive/Negative
<b>2. :TRIGger:SEQuence:SOURce</b>	
Syntax	:TRIGger:SEQuence:SOURce IMMEDIATE VIDeo EXTErnal :TRIGger:SEQuence:SOURce?
Function	Ses the trigger type. The query returns "IMM", "VID" or "EXT".
Default	IMMEDIATE
Front Panel	Trig, Trig Type, Free Run/Video/External
<b>3. :TRIGger:SEQuence:VIDeo:LEVel</b>	
Syntax	:TRIGger:SEQuence:VIDeo:LEVel <ampl> :TRIGger:SEQuence:VIDeo:LEVel?
Function	Sets the Video trigger level, the default unit is dBm, such as :TRIG:SEQ:VID:LEV 10 The query returns "1.000000E+01".
Explanations	<ul style="list-style-type: none"> <li>• &lt;ampl&gt; ranges from -300 dBm to 50 dBm.</li> <li>• Sending ":TRIGger:SEQuence:SOURce VIDeo" can select the Video trigger.</li> </ul>
Default	0 dBm
Front Panel	Trig, Trig Type, Video Trig Setup, Trig Level

## :UNIT

- :UNIT:POWer

1. :UNIT:POWer	
Syntax	:UNIT:POWer DBM DBMV DBUV V W :UNIT:POWer?
Function	Sets the unit of the input, output and the displayed amplitude. The query returns "DBM", "DBMV", "DBUV", "V" or "W".
Default	Log unit: dBm Lin unit: V
Front Panel	AMPT, Units, dBm/ dBmV/dBuV/Volts/Watts



## Chapter 3 Programming Examples

This chapter lists some programming examples in the development environments of Visual C++ 6.0 and Visual Basic 6.0 and LabVIEW 8.6 as well as programming example of how to control the analyzer to realize common functions in Linux operation system. All examples are based on VISA (Virtual Instrument Software Architecture).

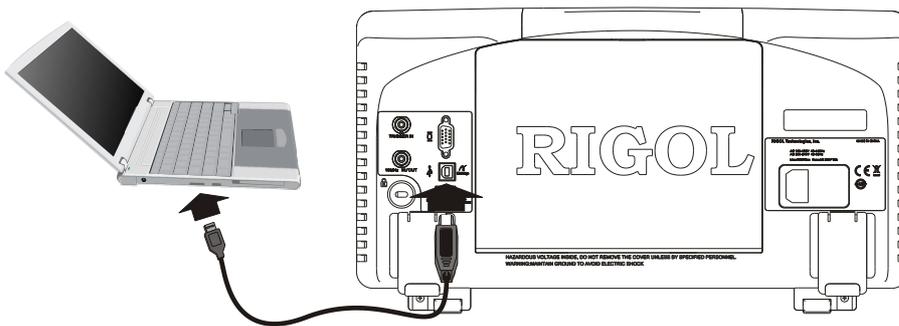
VISA is an API (Application Programming Interface) for controlling instruments. It is convenient for users to develop testing applications which are independent of the types of instrument and interface. Note the "VISA" mentioned here is NI (National Instrument)-VISA. NI-VISA is an API written by NI based on VISA standards. You can use NI-VISA to make a communication between the analyzer and the PC using USB or such instrument bus. As VISA has defined a set of software commands, users can control the instrument without understanding the working state of the interface bus. For more details, please refer to NI-VISA.

## Prepare for Programming

The programming preparations introduced below are only applicable to programming using Visual C++ 6.0, Visual Basic 6.0 and LabVIEW 8.6 in Windows operation system. For programming preparations in Linux operation system, please refer to "Linux Programming Example".

First make sure your computer has installed VISA library of NI (see <http://www.ni.com>). Here we install it in the default path: C:\Program Files\IVI Foundation\VISA.

Use USB cable to connect the USB Device interface at the rear panel of the analyzer with the USB interface of the PC for communication. See figure below.



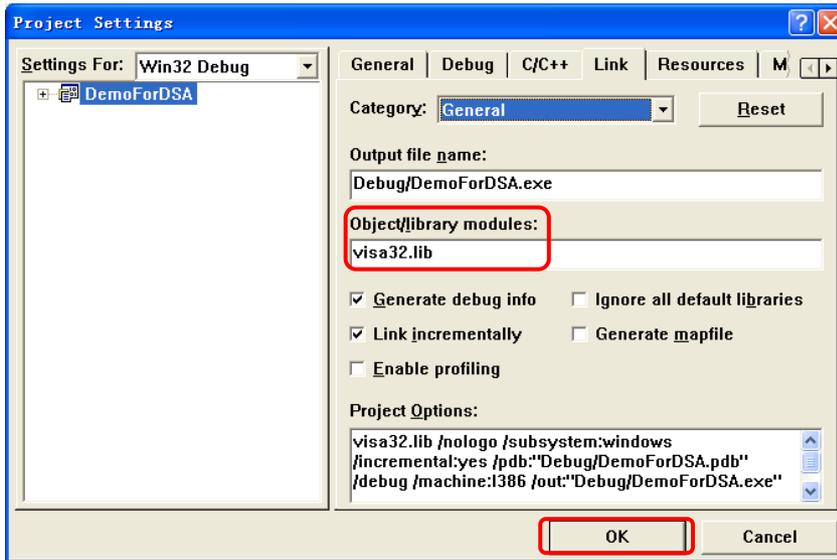
After successful connection, turn on the instrument, a dialog will guide you to install the driver of "USB Test and Measurement Device" on the PC. See figure below.



# Program in Visual C++ 6.0

Open Visual C++ 6.0, take the following steps:

1. Create a project based on MFC.
2. Click **Project**→**Settings**→**Link** and manually add the file **"visa32.lib"** in the **Object/library modules**.

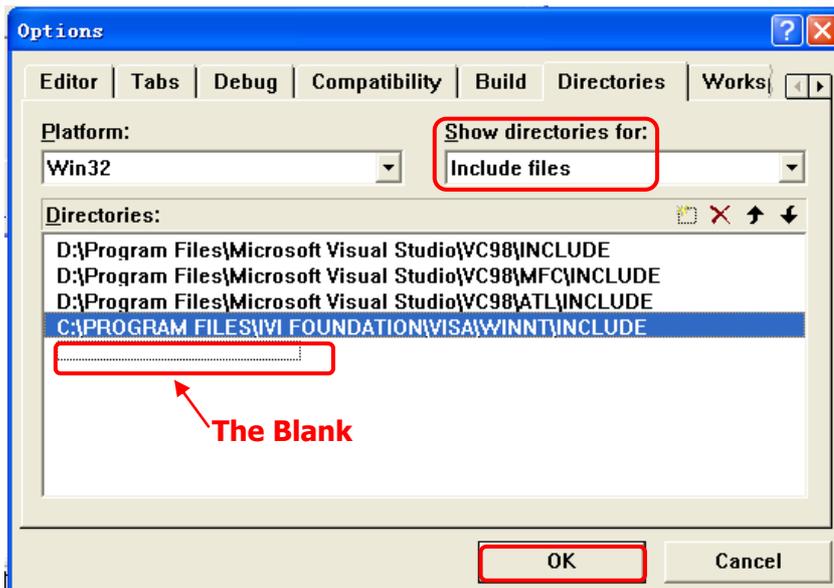


3. Choose **Tools**→**Options**→**Directories**.  
Select **"Include files"** in **Show directories for**, and then dblclick the blank in **Directories** to add the path of **"Include"**:

C:\Program Files\IVI Foundation\VISA\WinNT\include.

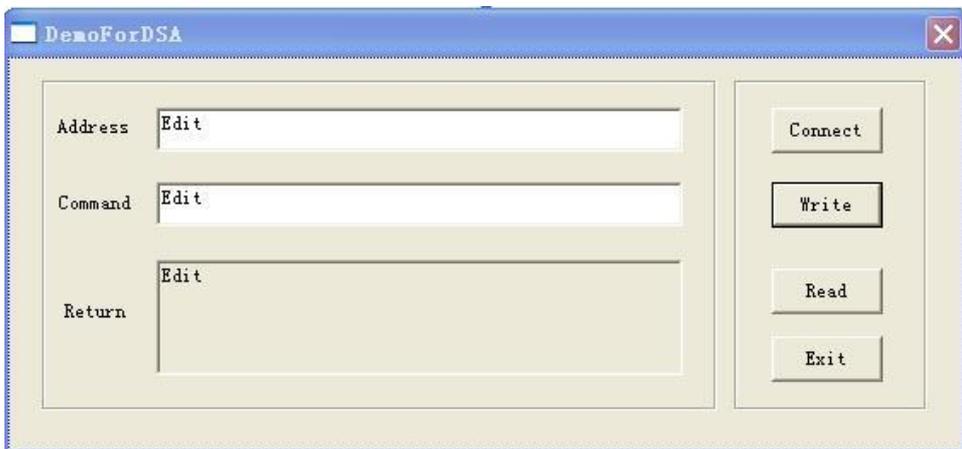
Select **"Library files"** in **Show directories for**, and then dblclick the blank in **Directories** to add the path of **"Lib"**:

C:\Program Files\IVI Foundation\VISA\WinNT\lib\msc.



**Note: At present, VISA library has been added successfully.**

4. Add controls: Text, Edit and Button. See figure below.



5. Add control variables.

Open **View**→**ClassWizard**→**Member Variables**, and add the following three variables:

Instrument address: CString m\_strInstrAddr

Command: CString m\_strCommand

Return Value CString m\_strResult

**6.** Encapsulate the read and write operations of VISA.

1) Encapsulate the write to VISA.

```
bool CDemoForDSADlg::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 of VISA.

```
bool CDemoForDSADlg::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;
}

```

**7.** Add the control message in order to response to codes.

1) Connect the instrument

```

void CDemoForDSADlg::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 bFindDSA = 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 DSA series then jump out
    //from the loop
    strInstr.MakeUpper();
    if (strInstr.Find("DSA") >= 0)
    {
        bFindDSA = true;
        m_strInstrAddr = strSrc;
        break;
    }

    //Find next instrument
    status = viFindNext(*findList,instrDesc);
}

if (bFindDSA == false)
{
    MessageBox("Didn't find any DSA!");
}
UpdateData(false);
}

```

## 2) Write operation

```

void CDemoForDSADlg::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

```

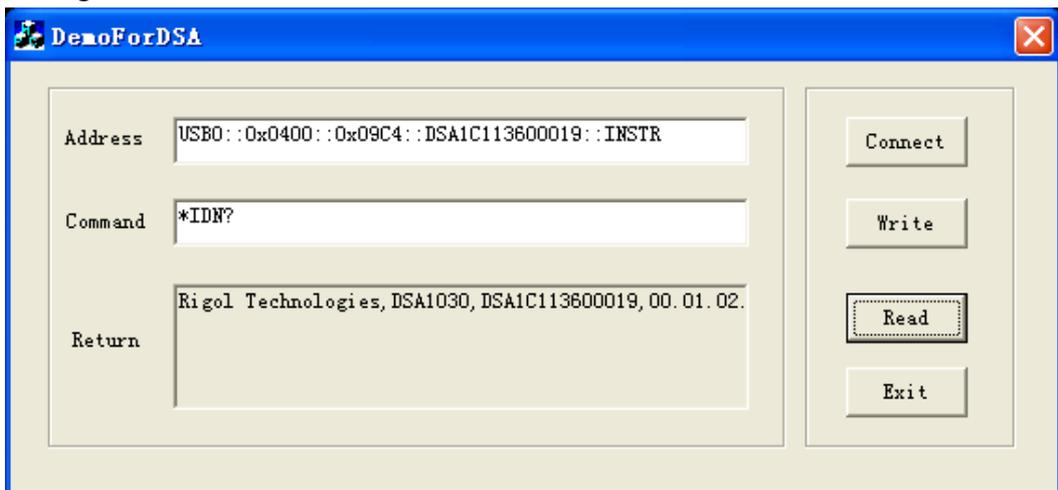
void CDemoForDSADlg::OnBtRead()           //Read operation
{
    // TODO: Add your control notification handler code here
    UpdateData(true);
    InstrRead(m_strInstrAddr,&m_strResult);
    UpdateData(false);
}

```

**8.** Execute results.

- 1) Click "Connect" to search the available analyzer.
- 2) Enter "\*IDN?" in the Command edit box.
- 3) Click "Write" to write the command into the analyzer.
- 4) Click "Read" to read the Return Value.

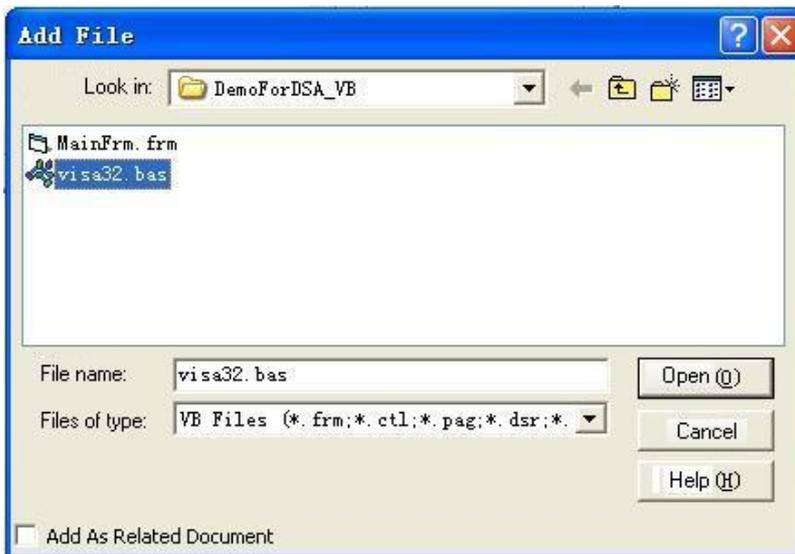
See figure below:



## Program in Visual Basic 6.0

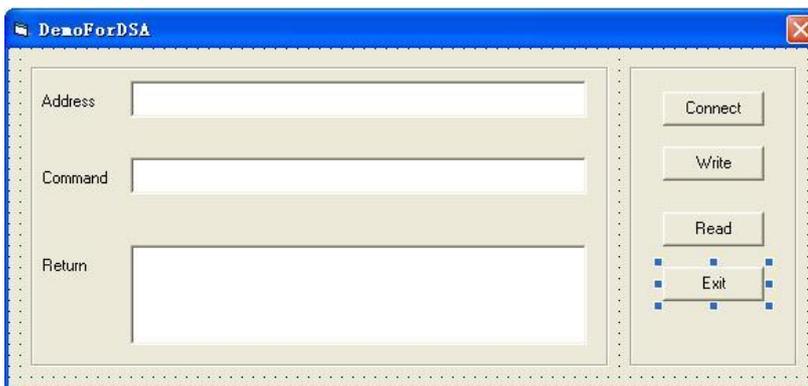
Open Visual Basic 6.0, take the following steps:

1. Create a **Standard EXE** project.
2. Choose **Project**→**Add File...** and add the file "**visa32.bas**" that contains all VISA functions and constant statements.



Then add **Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)** statement into the **visa32.bas**, or create a new module to declare the **Sleep** function.

3. Add controls: Text, Edit and Button, see figure below:



4. Encapsulate the read and write operations of VISA.  
 1) Encapsulate the write to VISA.

```
'-----
'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 " Failed to write to the instrument! "
        Exit Sub
    End If

    'Close the system
```

```

status = viClose(sesn)
status = viClose(dfltRM)

```

End Sub

2) Encapsulate the read of VISA.

```

'-----
'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)
    'Initial 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)
    'Open instrument failed
    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)
    ' Read 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 control event codes

1) Connect 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 bFindDSA As Boolean

```

```

'Initialize the system

```

```

status = viOpenDefaultRM(dfltRM)

```

```

' Initialize 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 instrument(resource)
strRet = ""
bFindDSA = False
For i = 0 To nList - 1
    ' Get the instrument name
    InstrWrite rsrcName(i), "*IDN?"
    Sleep 200
    strRet = InstrRead(rsrcName(i))
    ' Continue to switch the resource until find a DSA instrument
    strRet = UCase(strRet)
    j = InStr(strRet, "DSA")
    If (j >= 0) Then
        bFindDSA = True
        Exit For
    End If

    Call viFindNext(fList + i - 1, rsrcName(i))
Next i
'Display
If (bFindDSA = 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) Reas operation

'Read the return value from the instrument

```
Private Sub CmdRead_Click()
```

```
    Dim strTemp As String
```

```
    strTemp = InstrRead(TxtInsAddr.Text)
```

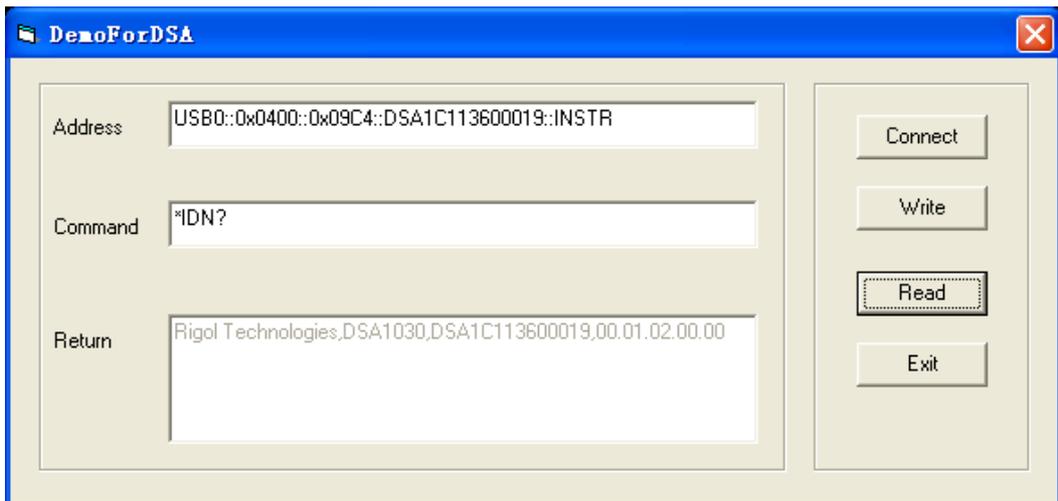
```
    TxtReturn.Text = strTemp
```

```
End Sub
```

## 6. Execute results

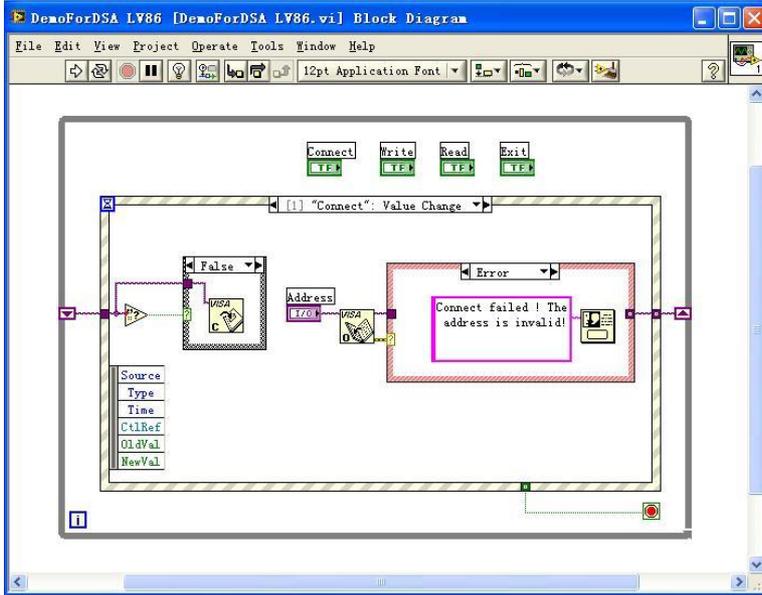
- 1) Click "Connect" to search the available analyzer.
- 2) Enter "\*IDN?" in the Command edit box.
- 3) Click "Write" to write the command into the analyzer.
- 4) Click "Read" to read the Return Value.

See figure below:

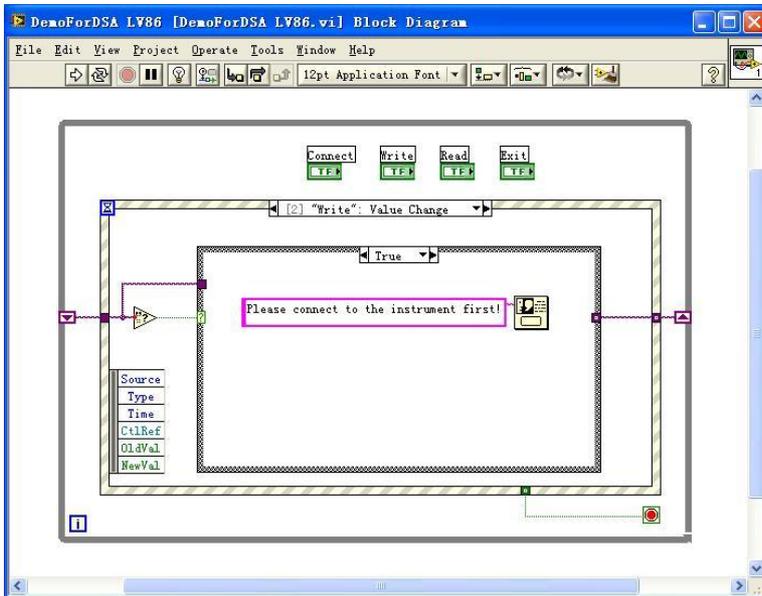


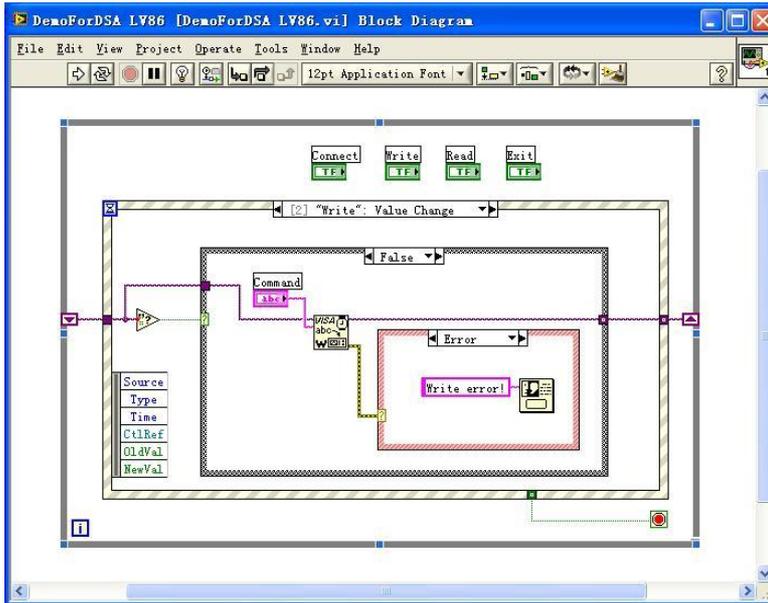


### 3. Instrument connection codes

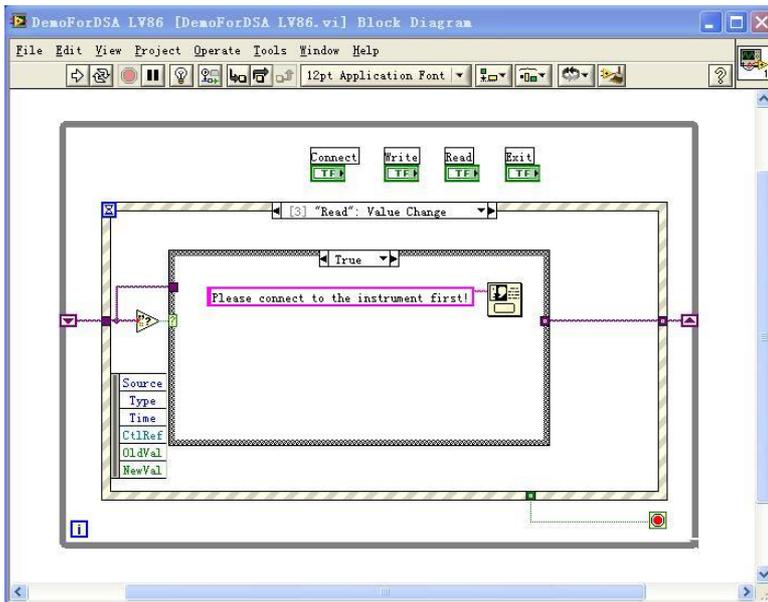


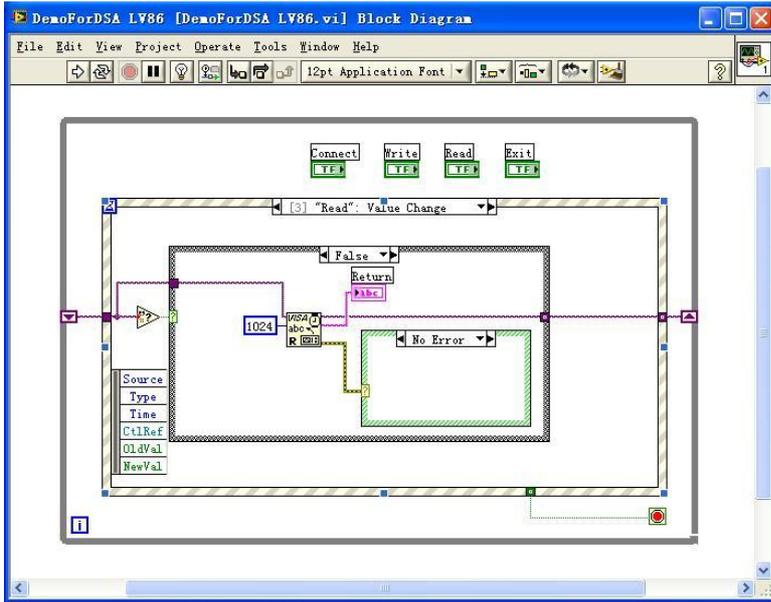
### 4. Write the operation codes (contain error judging)



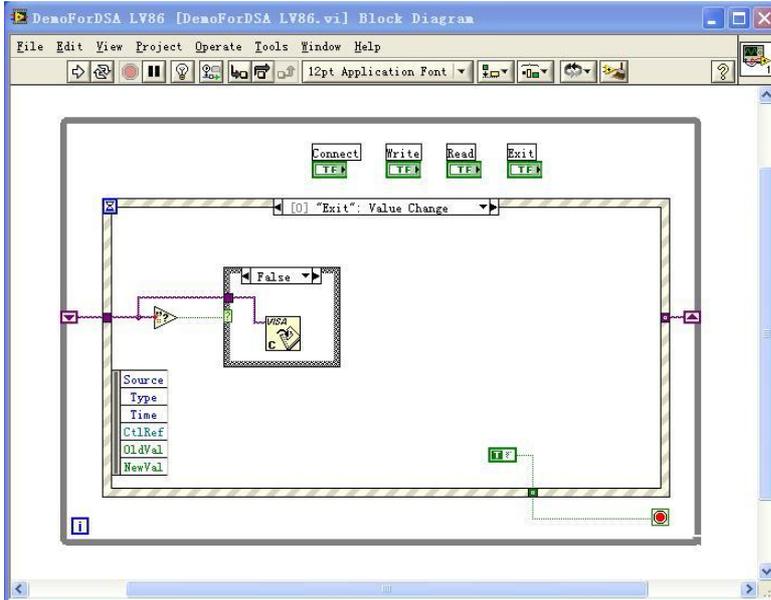


5. Read operation codes (contain error processing)

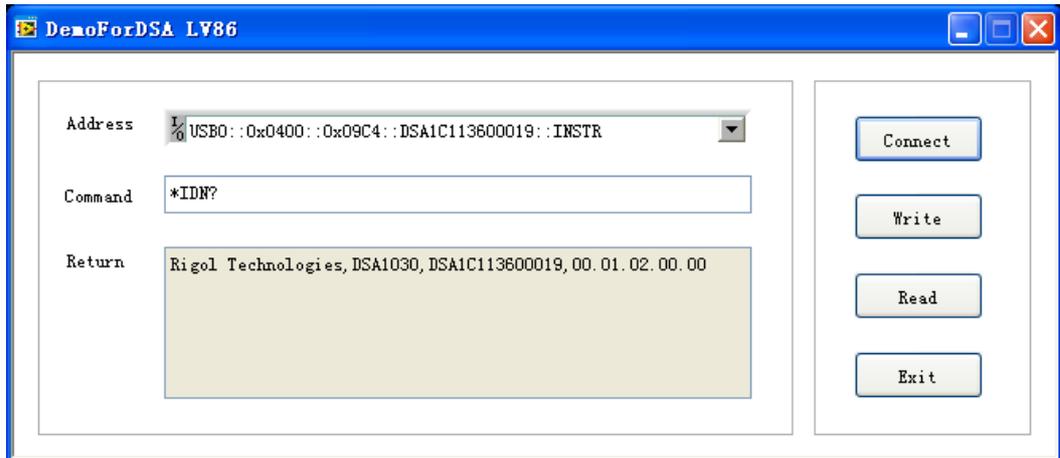




### 6. Exit code



## 7. Execute results

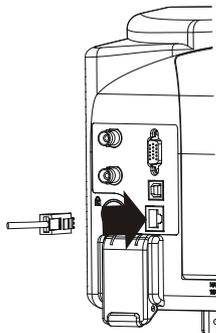


## Linux Programming Example

This section demonstrates how to use commands to control the analyzer to realize common functions in Linux operation system.

### Prepare for Programming

1. Programming environment used:  
Operation system: Fedroa 8 (Linux-2.6.23)  
GCC version: gcc-4.1.2
2. Install VISA library: make sure your computer has installed VISA library of NI (see <http://www.ni.com>). If not, install it following the steps below:  
First, download the VISA library (NI-VISA-4.4.0.ISO) from 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`
  - Unload the **iso** file  
`#umount NI_VISA`After the installation finishes, the default installation path is **/usr/local**.
3. As shown in the figure below, use network cable to connect the LAN interface at the rear panel of the analyzer with the LAN interface of the PC for communication. Users can also use network cable to connect the analyzer to the local network of the PC.



After the analyzer is correctly connected with the PC, configure the network address of the analyzer so that the address is in the same network segment with the PC network address. For example, the PC network address and DNS are set as follows, therefore, the network address of the analyzer should be set to:

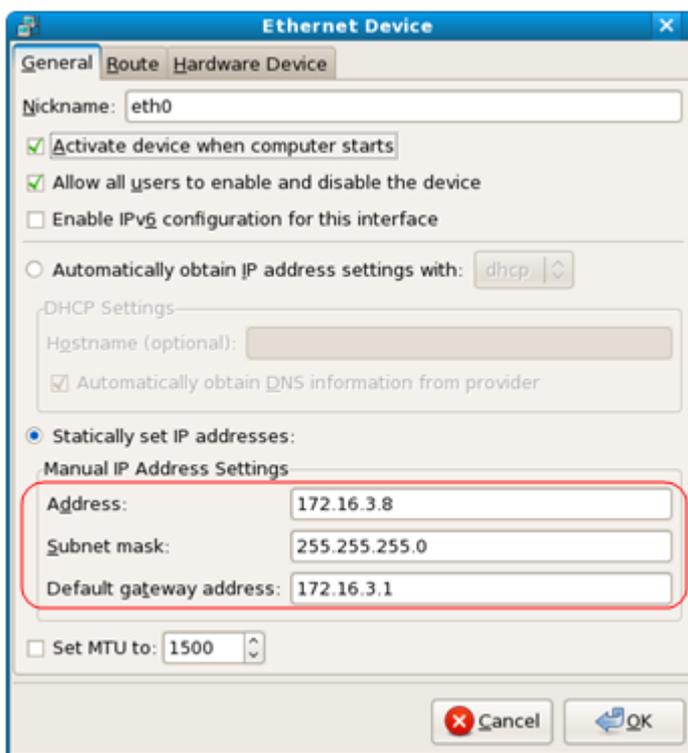
IP Address: 172.16.3.X\*

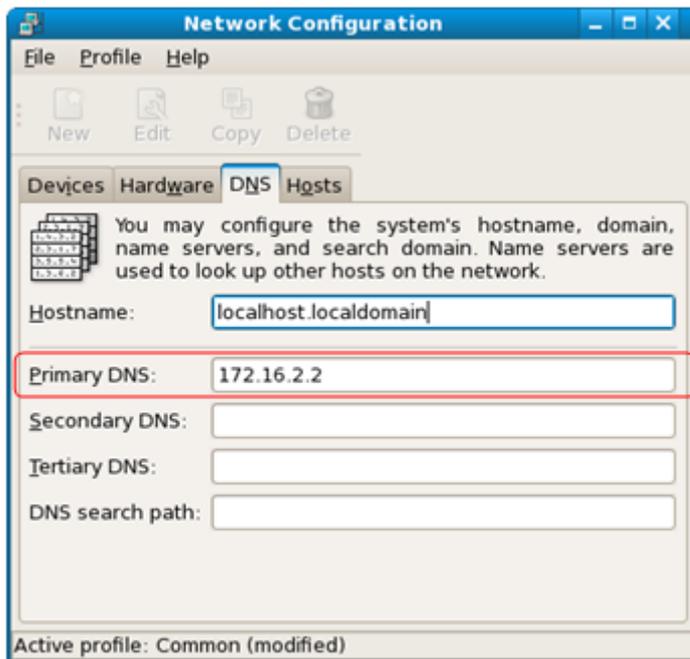
Gateway Address: 172.16.3.1

Subnet Mask: 255.255.255.0

DSN: 172.16.2.2

Note\*: X can be any value that has not been used between 2 and 254.



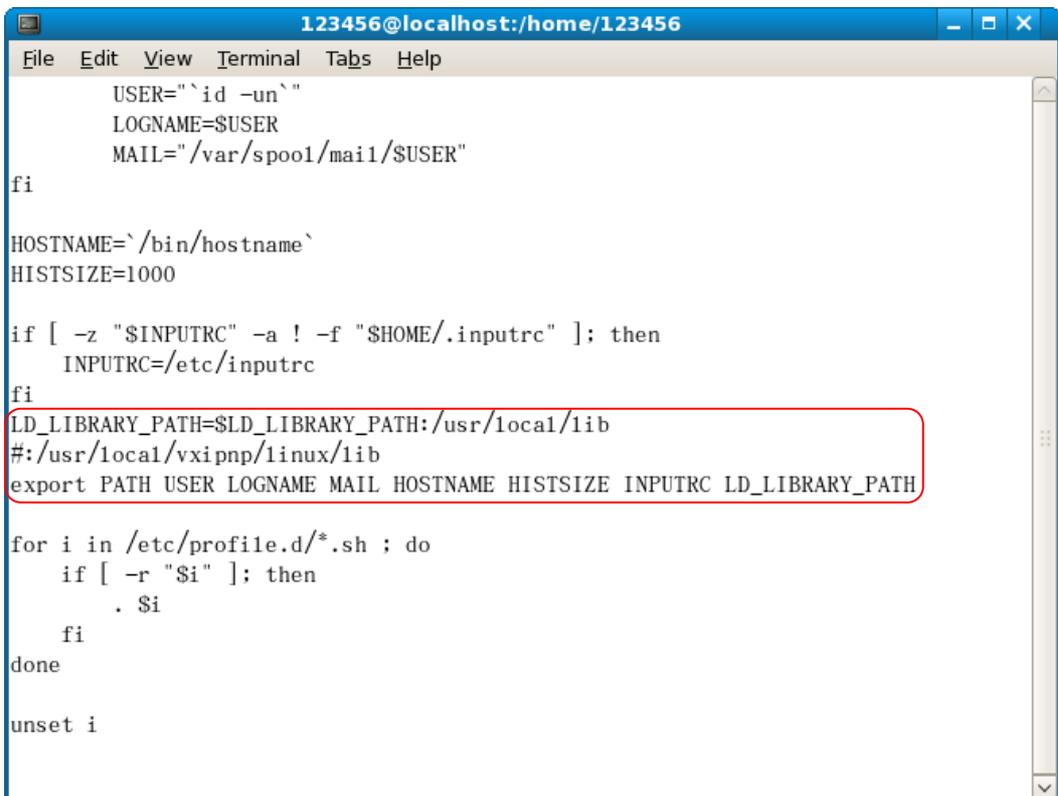


## Linux Programming Guide

First, using one of the two methods below to add the library location to the search path of the library so as to automatically load the installed library files.

Method 1: indicate 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.



```
123456@localhost:/home/123456
File Edit View Terminal Tabs Help
    USER="\`id -un`"
    LOGNAME=$USER
    MAIL="/var/spool/mail/$USER"
fi

HOSTNAME=`/bin/hostname`
HISTSIZE=1000

if [ -z "$INPUTRC" -a ! -f "$HOME/.inputrc" ]; then
    INPUTRC=/etc/inputrc
fi

LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/lib
#:/usr/local/vxipnp/linux/lib
export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE INPUTRC LD_LIBRARY_PATH

for i in /etc/profile.d/*.sh ; do
    if [ -r "$i" ]; then
        . $i
    fi
done

unset i
```

Method 2: add the search path of the library to the **/etc/ld.so.conf** file.

Adding 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**, execute the **/sbin/ldconfig** command to update **/etc/ld.so.cache** (this command must have root right) so as to locate the library when executing the program.



```

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 "DemoForDSA.h"

```

```

DemoForDSA::DemoForDSA()
{
    m_strInstrAddr = "";
    m_strResult    = "";
    m_strCommand   = "";
}

```

```

bool DemoForDSA::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 bFindDSA      = false;  
memset(instrDesc,0,1000);
```

```
//Open the VISA device
```

```
status = viOpenDefaultRM(&defaultRM);
```

```
if (status < VI_SUCCESS)  
{  
    cout<<"      No VISA equipment!"<<endl;  
    return false;  
}
```

```
//Search for resource
```

```
status = viFindRsrc(defaultRM,expr,findList, retcnt, instrDesc);
```

```
for (i = 0;i < (*retcnt);i++)  
{  
    //Get the instrument name  
    strSrc=instrDesc;  
  
    InstrWrite(strSrc, "*IDN?");  
    usleep(200);  
    InstrRead(strSrc,strInstr);
```

```
    //If it is a DSA series instrument, the program exits after finding the  
    instrument
```

```
    makeupper(strInstr);  
    if (strInstr.find("DSA",0) > 0)  
    {  
        bFindDSA = true;  
        m_strInstrAddr = strSrc;  
        break;  
    }
```

```
        //Get the next device□
        status = viFindNext(*findList,instrDesc);
    }

    if (bFindDSA == false)
    {
        printf("DSA device not found!\n");
        return false;
    }

    return true;
}

bool DemoForDSA::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 strig address to char* address
    SendAddr = const_cast<char*>(strAddr.c_str());

    //Address conversion, convert strig address to char* address
    SendBuf = const_cast<char*>(strContent.c_str());

    //Open the specific 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);

//Close the device
status = viClose(instr);
status = viClose(defaultRM);
return bWriteOK;
}

```

**bool** DemoForDSA::InstrRead(string strAddr, string & pstrResult) //Instrument Reads

```

{
    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 string address to char* address
    SendAddr=const_cast<char*>(strAddr.c_str());

    //Open the VISA device
    status = viOpenDefaultRM(&defaultRM);
    if (status < VI_SUCCESS)
    {
        // Error Initializing VISA...exiting
        cout<<"No VISA equipment!"<<endl;
        return false;
    }
}

```

```

        //Open the specific device□
        status = viOpen(defaultRM, SendAddr, VI_NULL, VI_NULL, &instr);

        //Read the device□
        status = viRead(instr, RecBuf, MAX_REC_SIZE, &retCount);

        //Close 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** and finish the flow control.

```

#include "DemoForDSA.h"

void menudisplay()
{
    cout<<"\t\t Please operate the instrument:\n          read  write
quit"<<endl;

```

```
}

int main()
{
    DemoForDSA 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 the instrument
        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 = DemoForDSA.cpp mainloop.cpp DemoForDSA.h
```

```
obj = DemoForDSA.o mainloop.o
```

```
INCLUDE= -I/usr/local/vxipnp/linux/include
```

```
LIB= -lvisa -lc -lpthread
```

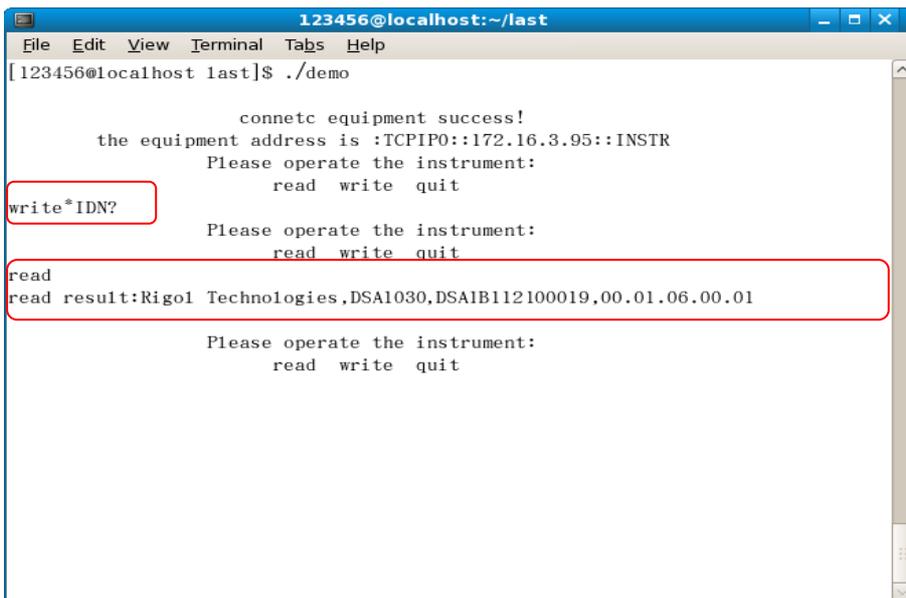
```
CC=g++
demo : $(obj)
    $(CC) $(INCLUDE) $(LIB) -o demo $(obj)

mainloop.o : mainloop.cpp DemoForDSA.h
    $(CC) -c $< -o $@
DemoForDSA.o: DemoForDSA.cpp DemoForDSA.h
    $(CC) -c $< -o $@

.PHONY : clean
clean:
    rm demo $(obj)
```

5. Execute results

- 1) #make
- 2) ./demo
- 3) Execute the program and connect the instrument automatically. If no instrument is found, "No VISA equipment!" is displayed and the program exits. If an instrument is found and successfully connected, the interface as shown in the figure below is displayed.
- 4) Input "write<command>" to write the command into the analyzer. For example, write<\*IDN?>.
- 5) Input "read" to read the return value as shown in the figure below.



# Command Quick Reference A-Z

*CLS	:CALCulate:MARKer:FCOunt:RESolution
*ESE	:CALCulate:MARKer:FCOunt:RESolution:AUTO
*ESE?	:CALCulate:MARKer:FCOunt[:STATe]
*ESR?	:CALCulate:MARKer:FCOunt:X?
*IDN?	:CALCulate:MARKer<n>:FUNction
*OPC	:CALCulate:MARKer<n>:MAXimum:MAX
*OPC?	:CALCulate:MARKer<n>:MAXimum:LEFT
*RST	:CALCulate:MARKer<n>:MAXimum:NEXT
*SRE	:CALCulate:MARKer<n>:MAXimum:RIGHT
*SRE?	:CALCulate:MARKer<n>:MINimum
*STB?	:CALCulate:MARKer<n>:MODE
*TRG	:CALCulate:MARKer<n>:PEAK:EXCursion
*WAI	:CALCulate:MARKer<n>:PEAK:SEARch:MODE
	:CALCulate:MARKer<n>:PEAK[:SET]:CF
	:CALCulate:MARKer<n>:PEAK:THReshold
	:CALCulate:MARKer<n>:PTPeak
	:CALCulate:MARKer<n>[:SET]:CENTer
	:CALCulate:MARKer<n>[:SET]:RLEVel
	:CALCulate:MARKer<n>[:SET]:START
	:CALCulate:MARKer<n>[:SET]:STEP
	:CALCulate:MARKer<n>[:SET]:STOP
	:CALCulate:MARKer<n>:STATe
	:CALCulate:MARKer:TABLE:STATe
	:CALCulate:MARKer<n>:TRACe
	:CALCulate:MARKer<n>:TRACe:AUTO
	:CALCulate:MARKer:TRCKing[:STATe]
	:CALCulate:MARKer<n>:X
	:CALCulate:MARKer<n>:X:CENTer
	:CALCulate:MARKer<n>:X:POSition
	:CALCulate:MARKer<n>:X:POSition:CENTer
	:CALCulate:MARKer<n>:X:POSition:SPAN
	:CALCulate:MARKer<n>:X:POSition:START
	:CALCulate:MARKer<n>:X:POSition:STOP
	:CALCulate:MARKer<n>:X:READout
	:CALCulate:MARKer<n>:X:SPAN
<b>A</b>	
:ABORT	
<b>C</b>	
:CALCulate:BANDwidth:NDB	
:CALCulate:BANDwidth:RESult?	
:CALCulate:LLINe:ALL:DELeTe	
:CALCulate:LLINe:CONTRol:DOMain	
:CALCulate:LLINe<n>:CONTRol:INTerpolate:TY PE	
:CALCulate:LLINe<n>:STATe	
:CALCulate:LLINe<n>:DATA	
:CALCulate:LLINe<n>:DATA:MERGe	
:CALCulate:LLINe<n>:DELeTe	
:CALCulate:LLINe:FAIL?	
:CALCulate:LLINe:FAIL:STOP:STATe	
:CALCulate:LLINe:FAIL:RATIo?	
:CALCulate:MARKer:AOff	
:CALCulate:MARKer<n>:CPEak[:STATe]	
:CALCulate:MARKer<n>:DELTA[:SET]:CENTer	
:CALCulate:MARKer<n>:DELTA[:SET]:SPAN	

:CALCulate:MARKer&lt;n&gt;:X:START

:CALCulate:MARKer&lt;n&gt;:X:STOP

:CALCulate:MARKer&lt;n&gt;:Y?

:CALCulate:NTData[:STATE]

:CALibration:[ALL]

:CALibration:AUTO

:CONFigure:ACPower

:CONFigure:CHPower

:CONFigure:CNRatio

:CONFigure:EBWidth

:CONFigure:HDISt

:CONFigure:OBWidth

:CONFigure:PF

:CONFigure:SANalyzer

:CONFigure:TOI

:CONFigure:TPOWer

:CONFigure?

:COUPLE

**D**

:DISPlay:AFUnction:POStion

:DISPlay:ANNotation:CLOCK

:DISPlay:BRIGtness

:DISPlay:ENABle

:DISPlay:MENU:STATe

:DISPlay:MENU:HTIME

:DISPlay:MSGswitch:STATe

:DISPlay:SKIN

:DISPlay:SSAVer:TIME

:DISPlay:WINdow:TRACe:GRATICule:GRID

:DISPlay:WINdow:TRACe:Y:DLINe

:DISPlay:WINdow:TRACe:Y:DLINe:STATe

:DISPlay:WINdow:TRACe:Y:SCALE:PDIVision

:DISPlay:WINdow:TRACe:Y:SCALE:RLEVEL

:DISPlay:WINdow:TRACe:Y:SCALE:RLEVEL:OFF

Set

:DISPlay:WINdow:TRACe:Y:SCALE:SPACing

:DISPlay:WINdow:TRACe:Y:SCALE:NRLevel

:DISPlay:WINdow:TRACe:Y:SCALE:NRPosition

**F**

:FETCh:ACPower?

:FETCh:ACPower:LOWer?

:FETCh:ACPower:UPPer?

:FETCh:ACPower:MAIN?

:FETCh:CHPower?

:FETCh:CHPower:CHPower?

:FETCh:CHPower:DENSity?

:FETCh:CNRatio?

:FETCh:CNRatio:CARRier?

:FETCh:CNRatio:CNRatio?

:FETCh:CNRatio:NOISe?

:FETCh:EBWidth?

:FETCh:HARMonics:AMPLitude:ALL?

:FETCh:HARMonics:AMPLitude? &lt;n&gt;

:FETCh:HARMonics[:DISTortion]?

:FETCh:HARMonics:FREQuency:ALL?

:FETCh:HARMonics:FREQuency? &lt;n&gt;

:FETCh:HARMonics:FUNDamental?

:FETCh:OBWidth?

:FETCh:OBWidth:OBWidth?

:FETCh:OBWidth:OBWidth:FERRor?

:FETCh:TOIntercept?

:FETCh:TOIntercept:IP3?

:FETCh:TPOWer?

:FORMat:BORDER

:FORMat[:TRACe][:DATA]

**H**

:HCOPY:ABORt

:HCOPY:IMAGe:COLor[:STATe]

:HCOPY:IMAGe:INVert

:HCOPY:IMAGe:PTIME

:HCOPY:IMAGe:QUALity

:HCOPY:IMAGe:FTYPE

:HCOPY[:IMMEDIATE]

:HCOPY:PAGE:ORIENTATION  
 :HCOPY:PAGE:PRINTS  
 :HCOPY:PAGE:SIZE  
 :HCOPY:RESUME

**I**

:INITIATE:CONTINUOUS  
 :INITIATE[:IMMEDIATE]  
 :INITIATE:PAUSE  
 :INITIATE:RESTART  
 :INITIATE:RESUME  
 :INPUT:IMPEDANCE

**M**

:MEMORY:DELETE  
 :MEMORY:DISK:FORMAT  
 :MEMORY:DISK:INFORMATION?  
 :MEMORY:MOVE  
 :MEMORY:LOAD:LIMIT  
 :MEMORY:LOAD:MTABLE  
 :MEMORY:LOAD:CORRECTION  
 :MEMORY:LOAD:STATE  
 :MEMORY:LOAD:TRACE  
 :MEMORY:LOAD:SETUP  
 :MEMORY:STORE:CORRECTION  
 :MEMORY:STORE:SCREEN  
 :MEMORY:STORE:STATE  
 :MEMORY:STORE:TRACE  
 :MEMORY:STORE:SETUP  
 :MEMORY:STORE:RESULTS  
 :MEMORY:STORE:MTABLE  
 :MEMORY:STORE:PTABLE  
 :MEMORY:STORE:LIMIT

**O**

:OUTPUT[:STATE]

**R**

:READ:ACPOWER?  
 :READ:ACPOWER:LOWER?  
 :READ:ACPOWER:UPPER?  
 :READ:ACPOWER:MAIN?  
 :READ:CHPOWER?  
 :READ:CHPOWER:CHPOWER?  
 :READ:CHPOWER:DENSITY?  
 :READ:CNRRATIO?  
 :READ:CNRRATIO:CARRIER?  
 :READ:CNRRATIO:CNRRATIO?  
 :READ:CNRRATIO:NOISE?  
 :READ:EBWIDTH?  
 :READ:HARMONICS:AMPLITUDE:ALL?  
 :READ:HARMONICS:AMPLITUDE? <n>  
 :READ:HARMONICS[:DISTORTION]?  
 :READ:HARMONICS:FREQUENCY:ALL?  
 :READ:HARMONICS:FREQUENCY? <n>  
 :READ:HARMONICS:FUNDAMENTAL?  
 :READ:OBWIDTH?  
 :READ:OBWIDTH:OBWIDTH?  
 :READ:OBWIDTH:OBWIDTH:FERROR?  
 :READ:TOINTERCEPT?  
 :READ:TOINTERCEPT:IP3?  
 :READ:TPOWER?

**S**

[:SENSE]:BANDWIDTH:RESOLUTION  
 [:SENSE]:BANDWIDTH:RESOLUTION:AUTO  
 [:SENSE]:BANDWIDTH:VIDEO  
 [:SENSE]:BANDWIDTH:VIDEO:AUTO  
 [:SENSE]:BANDWIDTH:VIDEO:RATIO  
 [:SENSE]:CORRECTION:CSET:ALL:DELETE  
 [:SENSE]:CORRECTION:CSET:ALL[:STATE]  
 [:SENSE]:CORRECTION:CSET<n>:DATA  
 [:SENSE]:CORRECTION:CSET<n>:DATA:MERGE  
 [:SENSE]:CORRECTION:CSET<n>:DELETE  
 [:SENSE]:CORRECTION:CSET<n>:X:SPACING  
 [:SENSE]:CORRECTION:CSET<n>[:STATE]

[[:SENSe]:CORRection:CSET:TABLE:STATE	[[:SENSe]:CHPower:AVERAge:COUNT
[[:SENSe]:DEMod	[[:SENSe]:CHPower:AVERAge[:STATE]
[[:SENSe]:DEMod:STATE	[[:SENSe]:CHPower:AVERAge:TCONtrol
[[:SENSe]:DEMod:TIME	[[:SENSe]:CHPower:BANDwidth:INTEgration
[[:SENSe]:DETEctor[:FUNcTION]	[[:SENSe]:CHPower:FREQuency:SPAN
[[:SENSe]:FREQuency:CENTer	[[:SENSe]:CNRatio:AVERAge:COUNT
[[:SENSe]:FREQuency:CENTer:SET:STEP	[[:SENSe]:CNRatio:AVERAge[:STATE]
[[:SENSe]:FREQuency:CENTer:STEP:AUTO	[[:SENSe]:CNRatio:AVERAge:TCONtrol
[[:SENSe]:FREQuency:CENTer:STEP[:INCRement]	[[:SENSe]:CNRatio:BANDwidth:INTEgration
[[:SENSe]:FREQuency:CENTer:UP	[[:SENSe]:CNRatio:BANDwidth:NOISE
[[:SENSe]:FREQuency:CENTer:DOWN	[[:SENSe]:CNRatio:OFFSet
[[:SENSe]:FREQuency:SPAN	[[:SENSe]:EBWidth:AVERAge:COUNT
[[:SENSe]:FREQuency:SPAN:FULL	[[:SENSe]:EBWidth:AVERAge[:STATE]
[[:SENSe]:FREQuency:SPAN:PREVIOUS	[[:SENSe]:EBWidth:AVERAge:TCONtrol
[[:SENSe]:FREQuency:SPAN:ZIN	[[:SENSe]:EBWidth:MAXHold:STATE
[[:SENSe]:FREQuency:SPAN:ZOUT	[[:SENSe]:EBWidth:FREQuency:SPAN
[[:SENSe]:FREQuency:START	[[:SENSe]:EBWidth:XDB
[[:SENSe]:FREQuency:STOP	[[:SENSe]:HDISt:AVERAge:COUNT
[[:SENSe]:POWer[:RF]:ATTenuation	[[:SENSe]:HDISt:AVERAge[:STATE]
[[:SENSe]:POWer[:RF]:ATTenuation:AUTO	[[:SENSe]:HDISt:AVERAge:TCONtrol
[[:SENSe]:POWer[:RF]:GAIN[:STATE]	[[:SENSe]:HDISt:NUMBers
[[:SENSe]:POWer[:RF]:MIXer:RANGE[:UPPer]	[[:SENSe]:HDISt:TIME
[[:SENSe]:POWer:ASCale	[[:SENSe]:HDISt:TIME:AUTO[:STATE]
[[:SENSe]:POWer:ARANge	[[:SENSe]:OBWidth:AVERAge:COUNT
[[:SENSe]:POWer:ATUNE	[[:SENSe]:OBWidth:AVERAge[:STATE]
[[:SENSe]:SWEep:POINts	[[:SENSe]:OBWidth:AVERAge:TCONtrol
[[:SENSe]:SWEep:COUNT	[[:SENSe]:OBWidth:MAXHold:STATE
[[:SENSe]:SWEep:COUNT:CURRENT?	[[:SENSe]:OBWidth:FREQuency:SPAN
[[:SENSe]:SWEep:TIME	[[:SENSe]:OBWidth:PERCent
[[:SENSe]:SWEep:TIME:AUTO	[[:SENSe]:STATus:ANALyzer?
[[:SENSe]:SWEep:TIME:AUTO:RULEs	[[:SENSe]:STATus:CORRections?
[[:SENSe]:ACPower:AVERAge:COUNT	[[:SENSe]:STATus:DEMod?
[[:SENSe]:ACPower:AVERAge[:STATE]	[[:SENSe]:STATus:POWer?
[[:SENSe]:ACPower:AVERAge:TCONtrol	[[:SENSe]:STATus:TG?
[[:SENSe]:ACPower:BANDwidth:INTEgration	[[:SENSe]:TOI:AVERAge:COUNT
[[:SENSe]:ACPower:BANDwidth:ACHannel	[[:SENSe]:TOI:AVERAge[:STATE]
[[:SENSe]:ACPower:CSPacing	[[:SENSe]:TOI:AVERAge:TCONtrol
	[[:SENSe]:TOI:FREQuency:SPAN

[:SENSe]:TPOWer:AVERAge:COUNT  
 [:SENSe]:TPOWer:AVERAge[:STATe]  
 [:SENSe]:TPOWer:AVERAge:TCONtrol  
 [:SENSe]:TPOWer:MODE  
 [:SENSe]:TPOWer:LLIMit  
 [:SENSe]:TPOWer:RLIMit  
 :SOURce:CORRection:OFFSet  
 :SOURce:FREQuency:OFFSet  
 :SOURce:POWer:LEVel:IMMEdiate:AMPLitude  
 :SOURce:POWer:MODE  
 :SOURce:POWer:SPAN  
 :SOURce:POWer:START  
 :SOURce:POWer:SWEep  
 :SOURce:TRACe:STORref  
 :SOURce:TRACe:REF:STATe  
 :STATus:OPERation:CONDition?  
 :STATus:OPERation:ENABLE  
 :STATus:OPERation[:EVENT]?  
 :STATus:PRESet  
 :STATus:QUEStionable:CONDition?  
 :STATus:QUEStionable:ENABLE  
 :STATus:QUEStionable[:EVENT]?  
 :SYSTem:COMMunicate:APORt  
 :SYSTem:COMMunicate:GPIB[:SELF]:ADDReSS  
 :SYSTem:COMMunicate:LAN[:SELF]:RESet  
 :SYSTem:COMMunicate:LAN[:SELF]:DHCP:STA  
 Te  
 :SYSTem:COMMunicate:LAN[:SELF]:AUTOip:ST  
 ATe  
 :SYSTem:COMMunicate:LAN[:SELF]:MANuip:S  
 TATe  
 :SYSTem:COMMunicate:LAN[:SELF]:IP:ADDReSS  
 s  
 :SYSTem:COMMunicate:LAN[:SELF]:IP:SUBMa  
 sk  
 :SYSTem:COMMunicate:LAN[:SELF]:IP:GATew  
 ay  
 :SYSTem:COMMunicate:LAN[:SELF]:IP:DNSSer  
 ver  
 :SYSTem:COMMunicate:USB[:SELF]:ADDReSS  
 :SYSTem:COMMunicate:USB[:SELF]:CLASs  
 :SYSTem:COMMunicate:USB[:SELF]:ADDReSS?  
 :SYSTem:COMMunicate:USB[:SELF]:CLASs  
 :SYSTem:CONFigure:INFomation?  
 :SYSTem:CONFigure:MESSAge?  
 :SYSTem:DATE  
 :SYSTem:ERRor[:NEXT]?  
 :SYSTem:FSWItch[:STATe]  
 :SYSTem:LANGUage  
 :SYSTem:LKEY  
 :SYSTem:OPTions?  
 :SYSTem:PON:TYPE  
 :SYSTem:PRESet  
 :SYSTem:PRESet:TYPE  
 :SYSTem:PRESet[:USER]:SAVE  
 :SYSTem:PROBe[:STATe]  
 :SYSTem:SPEaker[:STATe]  
 :SYSTem:TIME  
 :SYSTem:VERSIon?

**T**

:TRACe[:DATA]  
 :TRACe:MATH:PEAK[:DATA]?  
 :TRACe:MATH:PEAK:POINts?  
 :TRACe:MATH:PEAK:SORT  
 :TRACe:MATH:PEAK:TABLE:STATe  
 :TRACe:MATH:PEAK:THReShold  
 :TRACe:MATH:STATe  
 :TRACe:MATH:A  
 :TRACe:MATH:B  
 :TRACe:MATH:CONSt  
 :TRACe:MATH:TYPE  
 :TRACe<n>:MODE  
 :TRACe:AVERAge:CLear  
 :TRACe:AVERAge:COUNT  
 :TRACe:AVERAge:COUNT:CURRent?

:TRACe<n>:AVERAge:TYPE

:TRIGger:SEQuence:VIDeo:LEVel

:TRACe:CLEAr:ALL

:TRIGger:SEQuence:EXTErnal:SLOPe

## U

:TRIGger:SEQuence:SOURce

:UNIT:POWer