

RIGOL

Programming Guide

DS1000Z Series Digital Oscilloscope

Dec. 2013
RIGOL Technologies, Inc.

Guaranty and Declaration

Copyright

© 2013 RIGOL Technologies, Inc. All Rights Reserved.

Trademark Information

RIGOL is a registered trademark of RIGOL Technologies, Inc.

Publication Number

PGA17104-1110

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 be copied or photocopied or rearranged without prior written approval of **RIGOL**.

Product Certification

RIGOL guarantees this product conforms to the national and industrial standards in China as well as the ISO9001:2008 standard and the ISO14001:2004 standard. Other international standard conformance certification is in progress.

Contact Us

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

E-mail: service@rigol.com

Website: www.rigol.com

Document Overview

This manual guides users to control **RIGOL** DS1000Z series digital oscilloscope remotely by programming using SCPI commands through the remote interface. DS1000Z can build communication with the PC through the USB, LAN or GPIB (option) interface.

Main Topics in this Manual:

Chapter 1 Programming Overview

This chapter introduces how to build the remote communication between DS1000Z series digital oscilloscope and the PC. It also introduces the remote control method and the SCPI commands.

Chapter 2 Command System

This chapter introduces the syntax, function, parameter and using instruction of each command.

Chapter 3 Programming Demos

This chapter lists some programming demos to illustrate how to use commands to realize the common functions of the oscilloscope in the development environments of Excel, LabVIEW, Matlab, Visual Basic 6.0 and Visual C++ 6.0.

Tip
For the newest version of this manual, please download it from www.rigol.com.

Format Conventions in this Manual:

1. **Button**
The function key at the front panel is denoted by the format of "Button Name (Bold) + Text Box" in the manual, for example, **Utility** denotes the "Utility" key.
2. **Menu**
The menu item is denoted by the format of "Menu Word (Bold) + Character Shading" in the manual, for example, **System** denotes the "System" item under **Utility**.
3. **Operation Step**
The next step of the operation is denoted by an arrow "→" in the manual. For example, **Utility** → **System** denotes pressing **Utility** at the front panel and then pressing **System**.

Content Conventions in this Manual:

DS1000Z series includes the following models. Unless otherwise noted, this manual takes DS1104Z-S as an example to illustrate the command system of DS1000Z series.

Model	Analog Bandwidth	Channel	Source Channel
DS1104Z	100 MHz	4	None
DS1074Z	70 MHz	4	None
DS1104Z-S	100 MHz	4	2
DS1074Z-S	70 MHz	4	2

Contents

Guaranty and Declaration	I
Document Overview	II
Chapter 1 Programming Overview.....	1-1
To Build Remote Communication	1-2
Remote Control Methods	1-4
SCPI Command Overview	1-5
Syntax.....	1-5
Symbol Description	1-5
Parameter Type.....	1-6
Command Abbreviation	1-6
Chapter 2 Command System.....	2-1
:AUToscale	2-2
:CLEar.....	2-2
:RUN.....	2-2
:STOP	2-2
:SINGle	2-3
:TFORce.....	2-3
:ACQuire Commands.....	2-4
:ACQuire:AVERages.....	2-4
:ACQuire:MDEPth.....	2-5
:ACQuire:TYPE	2-6
:ACQuire:SRATe?.....	2-7
:CALibrate Commands.....	2-8
:CALibrate:QUIT	2-8
:CALibrate:START	2-8
:CHANnel<n> Commands	2-9
:CHANnel<n>:BWLimit.....	2-9
:CHANnel<n>:COUPling	2-10
:CHANnel<n>:DISPlay	2-10
:CHANnel<n>:INVert	2-10
:CHANnel<n>:OFFSet	2-11
:CHANnel<n>:RANGe	2-11
:CHANnel<n>:TCAL	2-12
:CHANnel<n>:SCALE	2-13
:CHANnel<n>:PROBe.....	2-13
:CHANnel<n>:UNITs.....	2-14
:CHANnel<n>:VERNier	2-14
:CURSor Commands	2-15
:CURSor:MODE	2-15
:CURSor:MANual	2-16
:CURSor:TRACk	2-21
:CURSor:AUTO:ITEM.....	2-25
:CURSor:XY	2-26
:DISPlay Commands	2-29
:DISPlay:CLEar	2-29
:DISPlay:DATA?	2-30
:DISPlay:TYPE	2-31
:DISPlay:GRADing:TIME	2-31
:DISPlay:WBRightness.....	2-31
:DISPlay:GRID	2-32
:DISPlay:GBRightness	2-32
IEEE488.2 Common Commands.....	2-33
*CLS	2-33
*ESE	2-33

*ESR?	2-34
*IDN?	2-34
*OPC	2-34
*RST	2-34
*SRE	2-34
*STB?	2-35
*TST?	2-35
*WAI	2-35
:MATH Commands	2-36
:MATH:DISPlay	2-37
:MATH:OPERator	2-37
:MATH:SOURce1	2-37
:MATH:SOURce2	2-38
:MATH:SCALe	2-38
:MATH:OFFSet	2-39
:MATH:INVert	2-39
:MATH:RESet	2-39
:MATH:FFT:WINDow	2-40
:MATH:FFT:SPLit	2-40
:MATH:FFT:UNIT	2-41
:MATH:FFT:HSCale	2-41
:MATH:FFT:HCENter	2-42
:MATH:OPTion:START	2-42
:MATH:OPTion:END	2-43
:MATH:OPTion:INVert	2-43
:MATH:OPTion:SENSitivity	2-44
:MATH:OPTion:DIStance	2-44
:MATH:OPTion:ASCale	2-44
:MATH:OPTion:THReshold1	2-45
:MATH:OPTion:THReshold2	2-45
:MASK Commands	2-46
:MASK:ENABle	2-46
:MASK:SOURce	2-47
:MASK:OPERate	2-47
:MASK:MDISplay	2-48
:MASK:SOOutput	2-48
:MASK:OUTPut	2-49
:MASK:X	2-49
:MASK:Y	2-49
:MASK:CREate	2-50
:MASK:PASSed?	2-50
:MASK:FAILed?	2-50
:MASK:TOTal?	2-50
:MASK:RESet	2-50
:MEASure Commands	2-51
:MEASure:SOURce	2-54
:MEASure:COUNter:SOURce	2-54
:MEASure:COUNter:VALue?	2-54
:MEASure:CLear	2-55
:MEASure:RECover	2-55
:MEASure:ADISplay	2-56
:MEASure:AMSource	2-56
:MEASure:SETup:MAX	2-57
:MEASure:SETup:MID	2-57
:MEASure:SETup:MIN	2-58
:MEASure:SETup:PSA	2-58
:MEASure:SETup:PSB	2-59
:MEASure:SETup:DSA	2-59

:MEASure:SETup:DSB	2-59
:MEASure:STATistic:DISPlay	2-60
:MEASure:STATistic:MODE	2-60
:MEASure:STATistic:RESet	2-61
:MEASure:STATistic:ITEM	2-61
:MEASure:ITEM	2-62
:REFeRence Commands	2-63
:REFeRence:DISPlay	2-63
:REFeRence<n>:ENABle	2-63
:REFeRence<n>:SOURce	2-64
:REFeRence<n>:VSCale	2-64
:REFeRence<n>:VOFFset	2-65
:REFeRence<n>:RESet	2-65
[[:SOURce<n>]] Commands (DS1000Z-S Only)	2-66
[:SOURce<n>]:OUTPut[<n>][:STATe]	2-67
[:SOURce<n>]:OUTPut[<n>]:IMPedance	2-67
[:SOURce<n>]:FREQuency[:FIXed]	2-68
[:SOURce<n>]:PHASe[:ADJust]	2-68
[:SOURce<n>]:PHASe:INITiate	2-69
[:SOURce<n>]:FUNCTion[:SHAPE]	2-69
[:SOURce<n>]:FUNCTion:RAMP:SYMMetry	2-70
[:SOURce<n>]:VOLTage[:LEVel][:IMMEDIATE][:AMPLitude]	2-70
[:SOURce<n>]:VOLTage[:LEVel][:IMMEDIATE]:OFFSet	2-71
[:SOURce<n>]:PULSe:DCYCLE	2-71
[:SOURce<n>]:MOD[:STATe]	2-72
[:SOURce<n>]:MOD:TYPE	2-72
[:SOURce<n>]:MOD:AM[:DEPTh]	2-73
[:SOURce<n>]:MOD:AM:INTernal:FREQuency	2-73
[:SOURce<n>]:MOD:FM:INTernal:FREQuency	2-73
[:SOURce<n>]:MOD:AM:INTernal:FUNCTion	2-74
[:SOURce<n>]:MOD:FM:INTernal:FUNCTion	2-74
[:SOURce<n>]:MOD:FM[:DEVlation]	2-74
[:SOURce<n>]:APPLy?	2-75
[:SOURce<n>]:APPLy:NOISe	2-75
[:SOURce<n>]:APPLy:PULSe	2-75
[:SOURce<n>]:APPLy:RAMP	2-75
[:SOURce<n>]:APPLy:SINusoid	2-75
[:SOURce<n>]:APPLy:SQUare	2-75
[:SOURce<n>]:APPLy:USER	2-75
:SYSTem Commands	2-77
:SYSTem:AUToscale	2-77
:SYSTem:BEEPer	2-77
:SYSTem:ERRor[:NEXT]?	2-78
:SYSTem:GPIB	2-78
:SYSTem:LANGuage	2-78
:SYSTem:LOCKed	2-78
:SYSTem:PON	2-79
:SYSTem:OPTion:INSTall	2-79
:SYSTem:OPTion:UNINSTall	2-79
:TIMebase Commands	2-80
:TIMebase:DELAy:ENABle	2-80
:TIMebase:DELAy:OFFSet	2-80
:TIMebase:DELAy:SCALE	2-81
:TIMebase[:MAIN]:OFFSet	2-81
:TIMebase[:MAIN]:SCALE	2-82
:TIMebase:MODE	2-82
:TRIGger Commands	2-83
:TRIGger:MODE	2-83

:TRIGger:COUPling	2-84
:TRIGger:STATus?	2-84
:TRIGger:SWEEp	2-84
:TRIGger:HOLDoff	2-85
:TRIGger:NREJect	2-85
:TRIGger:EDGe	2-86
:TRIGger:PULSe	2-88
:TRIGger:SLOPe	2-91
:TRIGger:VIDeo	2-96
:TRIGger:PATtern	2-99
:TRIGger:DURATIOn	2-101
:TRIGger:TIMEout (Option)	2-104
:TRIGger:RUNT (Option)	2-106
:TRIGger:WINDows (Option)	2-110
:TRIGger:DELAy (Option)	2-113
:TRIGger:SHOLd (Option)	2-116
:TRIGger:NEDGE (Option)	2-119
:TRIGger:RS232 (Option)	2-121
:TRIGger:IIC (Option)	2-125
:TRIGger:SPI (Option)	2-129
:WAVEform Commands	2-133
:WAVEform:SOURce	2-134
:WAVEform:MODE	2-135
:WAVEform:FORMat	2-135
:WAVEform:DATA?	2-136
:WAVEform:XINCrement?	2-137
:WAVEform:XORigin?	2-137
:WAVEform:XREFerence?	2-137
:WAVEform:YINCrement?	2-138
:WAVEform:YORigin?	2-138
:WAVEform:YREFerence?	2-138
:WAVEform:START	2-139
:WAVEform:STOP	2-139
:WAVEform:PREamble?	2-140
Chapter 3 Programming Demos	3-1
Programming Preparations	3-2
Excel Programming Demo	3-3
Matlab Programming Demo	3-7
LabVIEW Programming Demo	3-9
Visual Basic Programming Demo	3-13
Visual C++ Programming Demo	3-15

Chapter 1 Programming Overview

This chapter introduces how to build the remote communication between the PC and instrument and provides an overview of the syntax, symbol, parameter type and abbreviation rules of the SCPI commands.

Main topics of this chapter:

- ◆ [To Build Remote Communication](#)
- ◆ [Remote Control Methods](#)
- ◆ [SCPI Command Overview](#)

To Build Remote Communication

This oscilloscope can build communication with the PC through the USB, LAN or GPIB (option) interface. This section introduces how to control the oscilloscope remotely through the USB interface using Ultra Sigma in details.

Operation Steps:

1. Install the Ultra Sigma common PC software

Download the Ultra Sigma common PC software from www.rigol.com or acquire it from the resource CD supplied with the accessories and install it according to the instructions.

2. Connect the instrument and PC and configure the interface parameters of the instrument

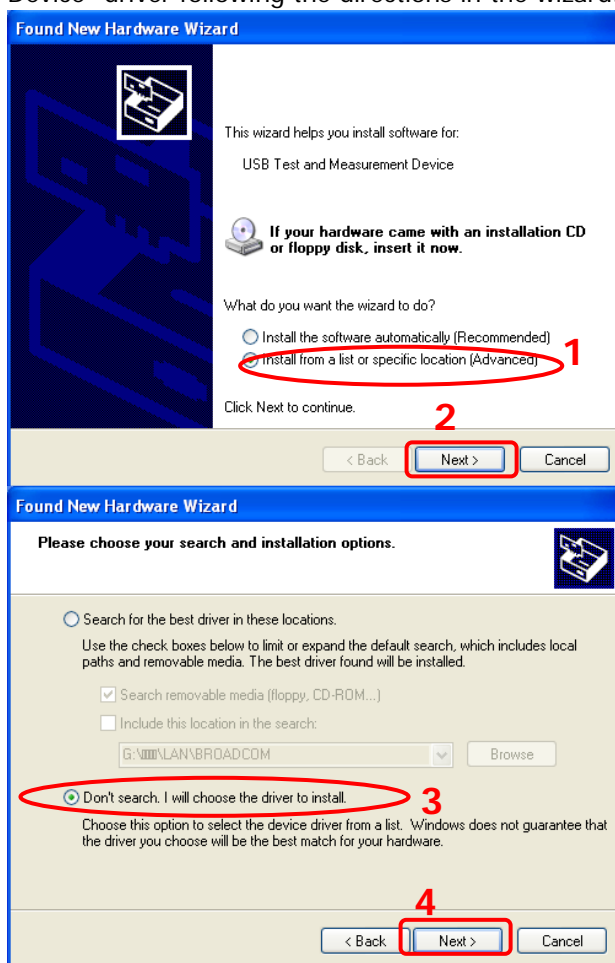
DS1000Z can build communication with the PC through the USB, LAN or GPIB (option) interface. This manual takes the USB interface as an example.

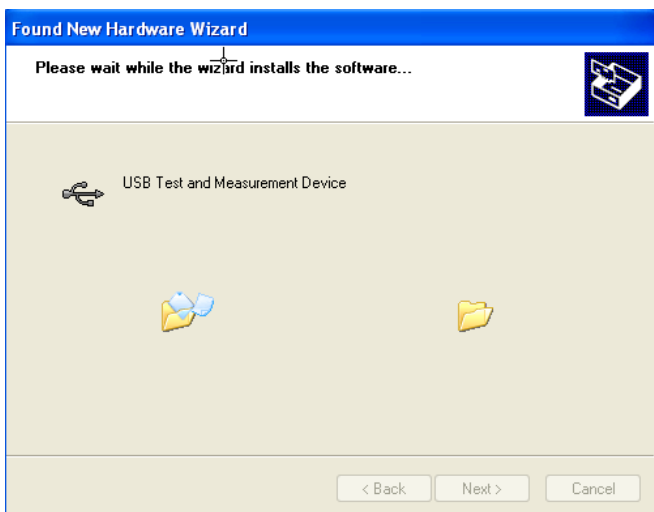
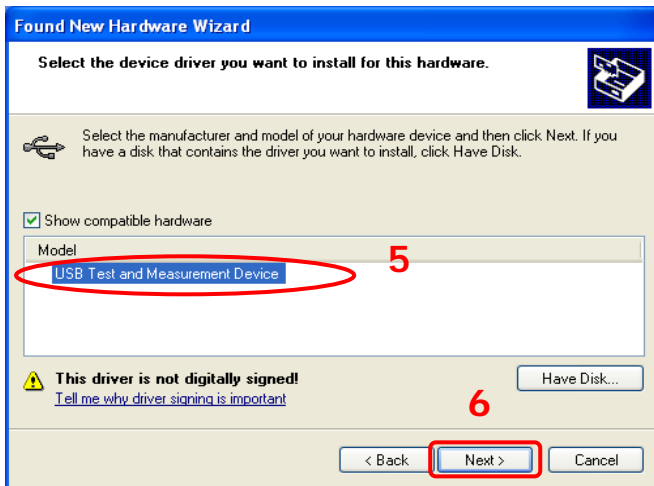
(1) Connect the devices

Connect the USB DEVICE interface at the real panel of the oscilloscope and the USB Host interface of the PC using a USB cable.

(2) Install the USB driver

This oscilloscope is a USBTMC device. Assuming that your PC has already been installed with **Ultra Sigma**, after you connect the oscilloscope to the PC and turn both on for the first time (the oscilloscope is automatically configured to the USB interface), the **New Hardware Wizard** as shown in the figure below is displayed on the PC. Please install the "USB Test and Measurement Device" driver following the directions in the wizard. The steps are as follows.





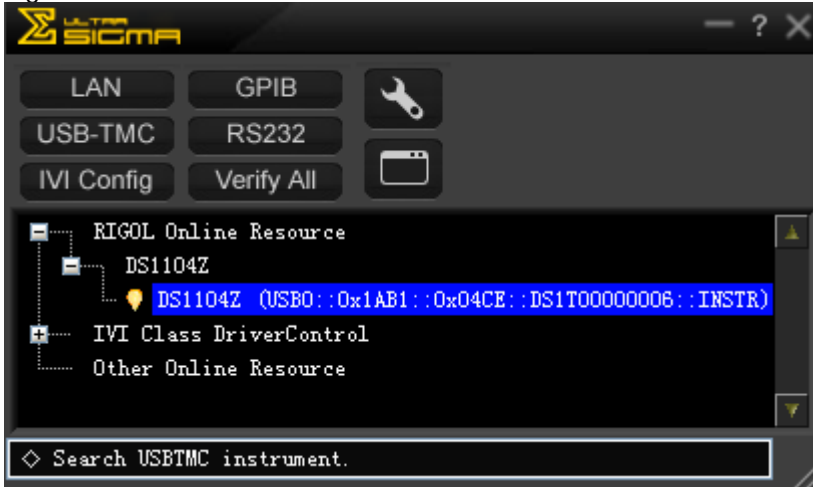
(3) Search for device resource

Start up the **Ultra Sigma** and the software will automatically search for the oscilloscope resources currently connected to the PC. You can also click **USB-TMC** to search for the resources. During the search, the status bar of the software is as shown in the figure below.



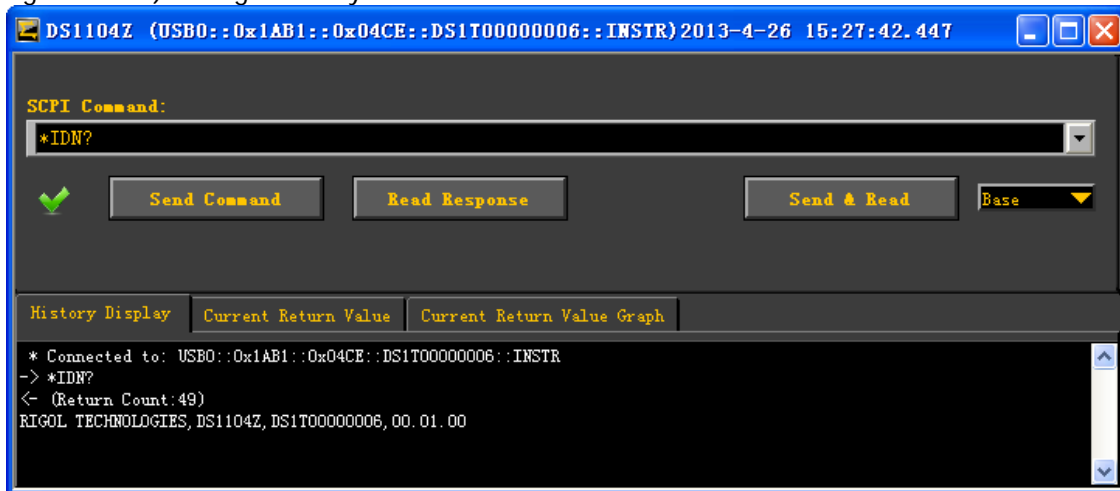
(4) View the device resources

The resources found will appear under the “RIGOL Online Resource” directory and the model number and USB interface information of the instrument will also be displayed as shown in the figure below.



(5) Communication test

Right click the resource name “DS1104Z (USB0::0x1AB1::0x04CE::DS1T00000006::INSTR)” and select “SCPI Panel Control” to turn on the remote command control panel (as shown in the figure below) through which you can send commands and read data.



Remote Control Methods

1. User-defined Programming

Users can use the SCPI (Standard Commands for Programmable Instruments) commands to program and control the oscilloscope. For details, refer to the introductions in [Chapter 3 Programming Demos](#).

2. Send SCPI Commands via the PC Software

You can control the oscilloscope remotely by sending SCPI commands via the PC software (Ultra Sigma) provided by **RIGOL**. Besides, you can also control the instrument using the “Measurement & Automation Explorer” of NI (National Instruments Corporation) or the “Agilent IO Libraries Suite” of Agilent (Agilent Technologies, Inc.).

SCPI Command Overview

SCPI (Standard Commands for Programmable Instruments) is a standardized instrument programming language that is built upon the standard IEEE488.1 and IEEE 488.2 and conforms to various standards (such as the floating point operation rule in IEEE754 standard, ISO646 7-bit coded character for information interchange (equivalent to ASCII programming)). The SCPI commands provide a hierarchical tree structure. Each command subsystem consists of a root keyword and on or more sub-keywords.

Syntax

The command string usually starts with ":"; the keywords are separated by ":" and are followed by the parameter settings available; "?" is added at the end of the command string to indicate query; the command and parameter are separated by space.

For example,
:ACQuire:TYPE <type>
:ACQuire:TYPE?

ACQuire is the root keyword of the command. TYPE is the second-level keyword. The command string starts with ":" which is also used to separate the multiple-level keywords. <type> represents the parameters available for setting. "?" represents query. The command :ACQuire:TYPE and parameter <type> are separated by space.

"," is generally used for separating multiple parameters contained in the same command, for example,
:TRIGger:PATtern:PATtern <pattern1>,<pattern2>,<pattern3>,<pattern4>

Symbol Description

The following symbols will not be sent with the commands.

1. Braces { }

The parameters enclosed in the braces are optional and are usually separated by the vertical bar "|". When using the command, one of the parameters must be selected.

2. Vertical Bar |

The vertical bar is used to separate multiple parameters and one of the parameters must be selected when using the command.

3. Square Brackets []

The content in the square brackets can be omitted.

4. Triangle Brackets < >

The parameter enclosed in the triangle brackets must be replaced by an effective value.

Parameter Type

1. Bool

The parameter could be OFF, ON, 0 or 1. For example,

```
:MEASure:ADISplay <bool>
```

```
:MEASure:ADISplay?
```

Wherein,

<bool> can be set to `{{0|OFF}}|{1|ON}}`.

The query returns 0 or 1.

2. Discrete

The parameter could be any of the values listed. For example,

```
:ACQuire:TYPE <type>
```

```
:ACQuire:TYPE?
```

Wherein,

<type> can be set to `NORMal|AVERages|PEAK|HRESolution`.

The query returns the abbreviations (NORM, AVER, PEAK or HRES).

3. Integer

Unless otherwise noted, the parameter can be any integer (NR1 format) within the effective value range. Note that do not set the parameter to a decimal, otherwise errors will occur. For example,

```
:DISPlay:GBrightness <brightness>
```

```
:DISPlay:GBrightness?
```

Wherein,

<brightness> can be set to any integer between 0 and 100.

The query returns an integer between 0 and 100.

4. Real

The parameter can be any real number within the effective value range and this command accepts decimal (NR2 format) and scientific notation (NR3 format) parameter input. For example,

```
:TRIGger:TIMEout:TIME <NR3>
```

```
:TRIGger:TIMEout:TIME?
```

Wherein,

<NR3> can be set to any real number between 1.600000e-08 (namely 16ns) to 0.1e+02 (namely 10s).

The query returns a real number in scientific notation.

5. ASCII String

The parameter should be the combinations of ASCII characters.

For example,

```
:SYSTem:OPTion:INSTall <license>
```

Wherein,

<license> can be set to `PDUY9N9QTS9PQSWPLAETRD3UJHYA`.

Command Abbreviation

All the commands are case-insensitive and you can use any of them. If abbreviation is used, all the capital letters in the command must be written completely. For example, the `:MEASure:ADISplay?` command can be abbreviated to `:MEAS:ADIS?`.

Chapter 2 Command System

This chapter introduces the syntax, function, parameter and using instruction of each DS1000Z command.

Main topics of this chapter:

- ◆ [:AUToscale](#)
- ◆ [:CLEar](#)
- ◆ [:RUN](#)
- ◆ [:STOP](#)
- ◆ [:SINGle](#)
- ◆ [:TFORce](#)
- ◆ [:ACQuire Commands](#)
- ◆ [:CALibrate Commands](#)
- ◆ [:CHANnel<n> Commands](#)
- ◆ [:CURSor Commands](#)
- ◆ [:DISPlay Commands](#)
- ◆ [IEEE488.2 Common Commands](#)
- ◆ [:MATH Commands](#)
- ◆ [:MASK Commands](#)
- ◆ [:MEASure Commands](#)
- ◆ [:REFerence Commands](#)
- ◆ [\[:SOURce\[<n>\]\] Commands \(DS1000Z-S Only\)](#)
- ◆ [:SYSTem Commands](#)
- ◆ [:TIMebase Commands](#)
- ◆ [:TRIGger Commands](#)
- ◆ [:WAVEform Commands](#)

Note:

1. Unless otherwise noted, this manual takes DS1104Z-S as an example to introduce the commands.
2. For parameter setting commands (for example, the time, frequency and amplitude), the oscilloscope can only accept numbers and set the parameters using the default units; it cannot recognize the units sent with the parameters. For the default unit of each parameter, please refer to the description in each command in the following introductions.

:AUToscale

Syntax :AUToscale

Description Enable the waveform auto setting function. The oscilloscope will automatically adjust the vertical scale, horizontal timebase and trigger mode according to the input signal to realize optimum waveform display. This command is equivalent to pressing the **AUTO** key at the front panel.

- Explanation**
- Theoretically, waveform auto setting function requires that the frequency of sine is no lower than 41Hz; the duty cycle should be greater than 1% and the amplitude must be at least 20mVpp for square.
 - When the pass/fail function is enabled (see the [:MASK:ENABLE](#) command), if you sent this command, the oscilloscope will disable the pass/fail function firstly and then execute the waveform auto setting function.
 - When the waveform record function is enabled or during the playback of the recorded waveform, this command is invalid.

:CLEar

Syntax :CLEar

Description Clear all the waveforms on the screen. If the oscilloscope is in the RUN state, waveform will still be displayed. This command is equivalent to pressing the **CLEAR** key at the front panel.

Related Command [:DISPlay:CLEar](#)

:RUN

:STOP

Syntax :RUN
:STOP

Description The :RUN command makes the oscilloscope start running and the :STOP command makes the oscilloscope stop running. These commands are equivalent to pressing the **RUN/STOP** key at the front panel.

Explanation When the waveform record function is enabled or during the playback of the recorded waveform, this command is invalid.

:SINGle

Syntax :SINGle

Description Set the oscilloscope to the single trigger mode. This command is equivalent to the following two operations: pressing the **SINGLE** key at the front panel and sending the [:TRIGger:SWEEp SINGle](#) command.

- Explanation**
- In the single trigger mode, the oscilloscope triggers once when the trigger conditions are met and then stops.
 - When the waveform record function is enabled or during the playback of the recorded waveform, this command is invalid.

Related Commands [:TFORce](#)
[:RUN](#)
[:STOP](#)

:TFORce

Syntax :TFORce

Description Generate a trigger signal forcefully. This command is only applicable to the normal and single trigger modes (see the [:TRIGger:SWEEp](#) command) and is equivalent to pressing the **FORCE** key at the front panel.

Related Command [:TRIGger:SWEEp](#)

:ACQUIRE Commands

The :ACQUIRE commands are used to set and query the memory depth, acquisition mode and the current sample rate of the oscilloscope as well as to set the number of averages under the average acquisition mode.

Command List ^[1]:

- ◆ [:ACQUIRE:AVERAGES](#)
- ◆ [:ACQUIRE:MDEPTH](#)
- ◆ [:ACQUIRE:TYPE](#)
- ◆ [:ACQUIRE:SRATE?](#)

Note^[1]: In the "Command List" in this manual, the parameters in the setting commands and the query commands are not included and you can refer to the complete introductions of the commands in the text according to the keyword.

:ACQUIRE:AVERAGES

Syntax :ACQUIRE:AVERAGES <count>

:ACQUIRE:AVERAGES?

Description Set or query the number of averages under the average acquisition mode.

Parameter

Name	Type	Range	Default
<count>	Integer	2 ⁿ (n is an integer from 1 to 10)	2

- Explanation**
- You can sent the [:ACQUIRE:TYPE](#) command to set the acquisition mode.
 - In the average acquisition mode, the greater the number of averages is, the lower the noise will be and the higher the vertical resolution will be but the slower the response of the displayed waveform to the waveform changes will be.

Return Format The query returns an integer between 2 and 1024.

Example :ACQUIRE:AVERAGES 128 /*Set the number of averages to 128*/
:ACQUIRE:AVERAGES? /*The query returns 128*/

Related Command [:ACQUIRE:TYPE](#)

:ACQUIRE:MDEPTH

Syntax :ACQUIRE:MDEPTH <mdep>

:ACQUIRE:MDEPTH?

Description Set or query the memory depth of the oscilloscope namely the number of waveform points that can be stored in a single trigger sample. The default unit is pts (points).

Parameter	Name	Type	Range	Default
	<mdep>	Discrete	Refer to the Explanation	AUTO

Explanation

- When a single channel is on: {AUTO|12000|120000|1200000|12000000|24000000}
- When dual channels are on: {AUTO|6000|60000|600000|6000000|12000000}
- When four channels are on: {AUTO|3000|30000|300000|3000000|6000000}
- Wherein, 24000000, 12000000 and 6000000 are options.
- The relationship among the memory depth, sample rate and waveform length satisfies the following equation:

$$\text{Memory Depth} = \text{Sample Rate} \times \text{Waveform Length}$$
 Wherein, the Waveform Length is the product of the horizontal timebase (set by the [:TIMEbase\[:MAIN\]:SCALE](#) command) times the number of the horizontal scales (12 for DS1000Z).
 When AUTO is selected, the oscilloscope will select the memory depth automatically according to the current sample rate.

Return Format The query returns the actual number of points (integer) or AUTO.

Example :ACQUIRE:MDEPTH 12000 /*Set the memory depth to 12000*/
 :ACQUIRE:MDEPTH? /*The query returns 12000*/

Related Commands [:ACQUIRE:SRATE?](#)
[:TIMEbase\[:MAIN\]:SCALE](#)

:ACQUIRE:TYPE

Syntax :ACQUIRE:TYPE <type>

:ACQUIRE:TYPE?

Description Set or query the acquisition mode when the oscilloscope samples.

Parameter	Name	Type	Range	Default
	<type>	Discrete	{NORMAl AVERages PEAK HRESolution}	NORMAl

- Explanation**
- **NORMAl**: in this mode, the oscilloscope samples the signal at equal time interval to rebuild the waveform. For most of the waveforms, the best display effect can be obtained using this mode.
 - **AVERages**: in this mode, the oscilloscope averages the waveforms from multiple samples to reduce the random noise of the input signal and improve the vertical resolution. The number of averages can be set by the [:ACQUIRE:AVERages](#) command. The greater the number of averages is, the lower the noise will be and the higher the vertical resolution will be but the slower the response of the displayed waveform to the waveform changes will be.
 - **PEAK (Peak Detect)**: in this mode, the oscilloscope acquires the maximum and minimum values of the signal within the sample interval to get the envelope of the signal or the narrow pulse of the signal that might be lost. In this mode, signal confusion can be prevented but the noise displayed would be larger.
 - **HRESolution (High Resolution)**: this mode uses a kind of ultra-sample technique to average the neighboring points of the sample waveform to reduce the random noise on the input signal and generate much smoother waveforms on the screen. This is generally used when the sample rate of the digital converter is higher than the storage rate of the acquisition memory.

Return Format The query returns NORM, AVER, PEAK or HRES.

Example :ACQUIRE:TYPE AVERages /*Select the average acquisition mode*/

Related Command [:ACQUIRE:AVERages](#)

:ACQUIRE:SRATE?

Syntax :ACQUIRE:SRATE?

Description Query the current sample rate. The default unit is Sa/s.

Explanation

- Sample rate is the sample frequency of the oscilloscope, namely the waveform points sampled per second.
- The relationship among the memory depth, sample rate and waveform length satisfies the following equation:
$$\text{Memory Depth} = \text{Sample Rate} \times \text{Waveform Length}$$
Wherein, the Memory Depth can be set using the [:ACQUIRE:MDEPTH](#) command, and the Waveform Length is the product of the horizontal timebase (set by the [:TIMEBASE\[:MAIN\]:SCALE](#) command) times the number of the horizontal scales (12 for DS1000Z).

Return Format The query returns the sample rate in scientific notation.

Example :ACQUIRE:SRATE? /*The query returns 2.000000e+09*/

Related Commands [:ACQUIRE:MDEPTH](#)
[:TIMEBASE\[:MAIN\]:SCALE](#)

:CALibrate Commands

Command List:

- ◆ [:CALibrate:QUIT](#)
- ◆ [:CALibrate:START](#)

:CALibrate:QUIT

Syntax :CALibrate:QUIT

Description Exit the calibration at any time.

Related Command [:CALibrate:START](#)

:CALibrate:START

Syntax :CALibrate:START

Description The oscilloscope starts to execute the self-calibration.

Explanation

- The self-calibration can make the oscilloscope quickly reach its optimum working state to obtain the most accurate measurement values.
- During the self-calibration, all the channels of the oscilloscope must be disconnected from the inputs.
- The functions of most of the keys are disabled during the self-calibration. You can send the [:CALibrate:QUIT](#) command to quit the self-calibration.

Related Command [:CALibrate:QUIT](#)

:CHANnel<n> Commands

The :CHANnel<n> commands are used to set or query the vertical system parameters, such as the bandwidth limit, coupling, vertical scale and vertical offset.

Command List:

- ◆ [:CHANnel<n>:BWLimit](#)
- ◆ [:CHANnel<n>:COUPLing](#)
- ◆ [:CHANnel<n>:DISPLay](#)
- ◆ [:CHANnel<n>:INVert](#)
- ◆ [:CHANnel<n>:OFFSet](#)
- ◆ [:CHANnel<n>:RANGe](#)
- ◆ [:CHANnel<n>:TCAL](#)
- ◆ [:CHANnel<n>:SCALe](#)
- ◆ [:CHANnel<n>:PROBe](#)
- ◆ [:CHANnel<n>:UNIIts](#)
- ◆ [:CHANnel<n>:VERNier](#)

:CHANnel<n>:BWLimit

Syntax :CHANnel<n>:BWLimit <type>

:CHANnel<n>:BWLimit?

Description Set or query the bandwidth limit parameter of the specified channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	--
<type>	Discrete	{20M OFF}	OFF

- Explanation**
- OFF: disable the bandwidth limit and the high frequency components of the signal under test can pass the channel.
20M: enable the bandwidth limit and the high frequency components that exceed 20 MHz are attenuated.
 - Enabling the bandwidth limit can reduce the noise, but can also attenuate the high frequency components.

Return Format The query returns 20M or OFF.

Example :CHANnel1:BWLimit 20M /*Enable the 20MHz bandwidth limit*/

:CHANnel<n>:COUPling**Syntax** :CHANnel<n>:COUPling <coupling>

:CHANnel<n>:COUPling?

Description Set or query the coupling mode of the specified channel.

Parameter	Name	Type	Range	Default
	<n>	Discrete	{1 2 3 4}	--
	<coupling>	Discrete	{AC DC GND}	DC

- Explanation**
- AC: the DC components of the signal under test are blocked.
 - DC: the DC and AC components of the signal under test can both pass the channel.
 - GND: the DC and AC components of the signal under test are both blocked.

Return Format The query returns AC, DC or GND.**Example** :CHANnel1:COUPling AC /*Select the AC coupling mode*/**:CHANnel<n>:DISPlay****Syntax** :CHANnel<n>:DISPlay <bool>

:CHANnel<n>:DISPlay?

Description Enable or disable the specified channel or query the status of the specified channel.

Parameter	Name	Type	Range	Default
	<n>	Discrete	{1 2 3 4}	--
	<bool>	Bool	{{0 OFF}} {1 ON}}	CH1: 1 ON Others: 0 OFF

Return Format The query returns 0 or 1.**Example** :CHANnel1:DISPlay ON /*Enable CH1*/**:CHANnel<n>:INVert****Syntax** :CHANnel<n>:INVert <bool>

:CHANnel<n>:INVert?

Description Enable or disable the inverted display mode of the specified channel or query the status of the inverted display mode of the specified channel.

Parameter	Name	Type	Range	Default
	<n>	Discrete	{1 2 3 4}	--
	<bool>	Bool	{{0 OFF}} {1 ON}}	0 OFF

Explanation When the inverted display mode is enabled, the oscilloscope displays the waveform inverted taking the vertical offset horizontal line as reference.**Return Format** The query returns 0 or 1.**Example** :CHANnel1:INVert ON /*Enable the inverted display mode of CH1*/

:CHANnel<n>:OFFSet

Syntax :CHANnel<n>:OFFSet <offset>

:CHANnel<n>:OFFSet?

Description Set or query the vertical offset of the specified channel. The default unit is V.

Parameter	Name	Type	Range	Default
	<n>	Discrete	{1 2 3 4}	--
	<offset>	Real	Related to the current vertical scale and probe ratio. When the probe ratio is 1, vertical scale \geq 500mV/div: -100V to +100V vertical scale<500mV/div: -2V to +2V	0V

Return Format The query returns the vertical offset in scientific notation.

Example :CHANnel1:OFFSet 0.01 /*Set the vertical offset of CH1 to 10mV*/
:CHANnel1:OFFSet? /*The query returns 1.000000e-02*/

:CHANnel<n>:RANGe

Syntax :CHANnel<n>:RANGe <range>

:CHANnel<n>:RANGe?

Description Set or query the vertical range of the specified channel. The default unit is V.

Parameter	Name	Type	Range	Default
	<n>	Discrete	{1 2 3 4}	--
	<range>	Real	Related to the current vertical scale and probe ratio. When the probe ratio is 1, 8mV to 80V	8V

Explanation This command indirectly modifies the vertical scale of the specified channel (vertical range=8*vertical scale). The vertical scale can be set by the [:CHANnel<n>:SCALE](#) command.

Return Format The query returns the vertical range in scientific notation.

Example :CHANnel1:RANGe 8 /*Set the vertical range of CH1 to 8V*/
:CHANnel1:RANGe? /*The query returns 8.000000e+00*/

Related Command [:CHANnel<n>:SCALE](#)

:CHANnel<n>:TCAL**Syntax** :CHANnel<n>:TCAL <val>

:CHANnel<n>:TCAL?

Description Set or query the delay calibration time of the specified channel to calibrate the zero offset of the corresponding channel. The default unit is s.

Parameter	Name	Type	Range	Default
	<n>	Discrete	{1 2 3 4}	--
	<val>	Real	-100ns to 100ns	0.00ns

Explanation <val> can only be set to the specific values in the specified step. If the parameter you sent is not one of the specific values, the parameter will be set to the nearest specific values automatically. The step varies with the horizontal timebase (set by the [:TIMebase\[:MAIN\]:SCALE](#) command).

Horizontal Timebase	Step of the Delay Calibration Time
5ns	100ps
10ns	200ps
20ns	400ps
50ns	1ns
100ns	2ns
200ns	4ns
500ns	10ns
1 μ s and above	20ns

Return Format The query returns the delay calibration time in scientific notation.**Example** :CHANnel1:TCAL 0.00002 /*Set the delay calibration time to 20ns*/
:CHANnel1:TCAL? /*The query returns2.000000e-05*/**Related Command** [:TIMebase\[:MAIN\]:SCALE](#)

:CHANnel<n>:SCALE

Syntax :CHANnel<n>:SCALE <scale>

:CHANnel<n>:SCALE?

Description Set or query the vertical scale of the specified channel. The default unit is V.

Parameter	Name	Type	Range	Default
	<n>	Discrete	{1 2 3 4}	--
	<scale>	Real	Related to the current probe ratio. probe ratio = 1: 1mV to 10V probe ratio = 10 (default): 10mV to 100V	1V (the probe ratio is 10)

- Explanation**
- The range of the vertical scale is related to the current probe ratio (set by the [:CHANnel<n>:PROBE](#) command). As the default probe ratio of DS1000Z is 10, the default value of <scale> is the value when the probe ratio is 10.
 - You can use the [:CHANnel<n>:VERNier](#) command to enable or disable the fine adjustment of the vertical scale. By default, the fine adjustment is off. At this point, you can only set the vertical scale in 1-2-5 step, namely 1mV, 2mV, 5mV, 10mV, ..., 10V. When the fine adjustment is on, you can further adjust the vertical scale within a relatively smaller range to improve the vertical resolution. If the amplitude of the input waveform is a little bit greater than the full scale under the current scale and the amplitude would be a little bit lower if the next scale is used, fine adjustment can be used to improve the display amplitude of the waveform to view the signal details.

Return Format The query returns the vertical scale in scientific notation.

Example :CHANnel1:SCALE 1 /*Set the vertical scale of CH1 to 1V*/
:CHANnel1:SCALE? /*The query returns 1.000000e+00*/

Related Command [:CHANnel<n>:PROBE](#)
[:CHANnel<n>:VERNier](#)

:CHANnel<n>:PROBE

Syntax :CHANnel<n>:PROBE <atten>

:CHANnel<n>:PROBE?

Description Set or query the probe ratio of the specified channel.

Parameter	Name	Type	Range	Default
	<n>	Discrete	{1 2 3 4}	--
	<atten>	Discrete	{0.01 0.02 0.05 0.1 0.2 0.5 1 2 5 10 20 50 100 200 500 1000}	10

- Explanation**
- Setting the probe ratio refers to multiply the signal sampled with the specified ratio and then display the result (the actual amplitude of the signal will be not affected).
 - Setting the probe ratio will affect the range of the vertical scale.

Return Format The query returns the probe ratio in scientific notation.

Example :CHANnel1:PROBE 10 /*Set the probe ratio to 10*/
:CHANnel1:PROBE? /*The query returns 1.000000e+01*/

Related Command [:CHANnel<n>:SCALE](#)

:CHANnel<n>:UNITs**Syntax** :CHANnel<n>:UNITs <units>

:CHANnel<n>:UNITs?

Description Set or query the amplitude display unit of the specified channel.

Parameter	Name	Type	Range	Default
	<n>	Discrete	{1 2 3 4}	--
	<units>	Discrete	{VOLTage WATT AMPere UNKNown}	VOLTage

Return Format The query returns VOLT, WATT, AMP or UNKN.**Example** :CHANnel1:UNITs VOLTage /*Set the amplitude display unit of CH1 to V*/**:CHANnel<n>:VERNier****Syntax** :CHANnel<n>:VERNier <bool>

:CHANnel<n>:VERNier?

Description Enable or disable the fine adjustment of the vertical scale of the specified channel, or query the fine adjustment status of the vertical scale of the specified channel.

Parameter	Name	Type	Range	Default
	<n>	Discrete	{1 2 3 4}	--
	<bool>	Bool	{{0 OFF}} {1 ON}}	0 OFF

Explanation By default, the fine adjustment is off. At this point, you can only set the vertical scale in 1-2-5 step, namely 1mV, 2mV, 5mV, 10mV...10V. When the fine adjustment is on, you can further adjust the vertical scale within a relatively smaller range to improve the vertical resolution. If the amplitude of the input waveform is a little bit greater than the full scale under the current scale and the amplitude would be a little bit lower if the next scale is used, fine adjustment can be used to improve the display amplitude of the waveform to view the signal details.**Return Format** The query returns 0 or 1.**Example** :CHANnel1:VERNier ON /*Enable the fine adjustment of the vertical scale of CH1*/**Related Command** [:CHANnel<n>:SCALE](#)

:CURSor Commands

The :CURSor commands are used to measure the X-axis value (such as time) and Y-axis value (such as voltage) of the waveform displayed on the screen.

Command List:

- ◆ [:CURSor:MODE](#)
- ◆ [:CURSor:MANual](#)
- ◆ [:CURSor:TRACk](#)
- ◆ [:CURSor:AUTO:ITEM](#)
- ◆ [:CURSor:XY](#)

:CURSor:MODE

Syntax :CURSor:MODE <mode>

:CURSor:MODE?

Description Set or query the cursor measurement mode.

Parameter

Name	Type	Range	Default
<mode>	Discrete	{OFF MANual TRACk AUTO XY}	OFF

Explanation

- OFF: disable the cursor measurement function.
- MANual: enable the manual cursor measurement mode.
- TRACk: enable the track cursor measurement mode.
- AUTO: enable the auto cursor measurement mode.
- XY: enable the XY cursor measurement mode. This mode is valid only when the horizontal timebase mode is XY.

Return Format

The query returns OFF, MAN, TRAC, AUTO or XY.

Example :CURSor:MODE MANual /*Enable the manual cursor measurement mode*/

Related Commands

[:CURSor:MANual](#)

[:CURSor:TRACk](#)

[:CURSor:XY](#)

[:TIMebase:MODE](#)

:CURSor:MANual

Command List:

- ◆ [:CURSor:MANual:TYPE](#)
- ◆ [:CURSor:MANual:SOURce](#)
- ◆ [:CURSor:MANual:TUNit](#)
- ◆ [:CURSor:MANual:VUNit](#)
- ◆ [:CURSor:MANual:AX](#)
- ◆ [:CURSor:MANual:BX](#)
- ◆ [:CURSor:MANual:AY](#)
- ◆ [:CURSor:MANual:BY](#)
- ◆ [:CURSor:MANual:AXValue?](#)
- ◆ [:CURSor:MANual:AYValue?](#)
- ◆ [:CURSor:MANual:BXValue?](#)
- ◆ [:CURSor:MANual:BYValue?](#)
- ◆ [:CURSor:MANual:XDELta?](#)
- ◆ [:CURSor:MANual:IXDELta?](#)
- ◆ [:CURSor:MANual:YDELta?](#)

:CURSor:MANual:TYPE

Syntax :CURSor:MANual:TYPE <type>
 :CURSor:MANual:TYPE?

Description Set or query the cursor type in manual cursor measurement mode.

Parameter	Name	Type	Range	Default
	<type>	Discrete	{X Y}	X

Explanation

- X: select the X type cursors. The X type cursors are a vertical solid line (cursor A) and a vertical dotted line (cursor B) and are usually used to measure the time parameters.
- Y: select the Y type cursors. The Y type cursors are a horizontal solid line (cursor A) and a horizontal dotted line (cursor B) and are usually used to measure the voltage parameters.

Return Format The query returns X or Y.

Example :CURSor:MANual:TYPE Y /*select the Y type cursors*/

:CURSor:MANual:SOURce

Syntax :CURSor:MANual:SOURce <source>

:CURSor:MANual:SOURce?

Description Set or query the channel source of the manual cursor measurement mode.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4 MATH}	CHANnel1

Explanation Only the channel that is enabled currently can be selected.

Return Format The query returns CHAN1, CHAN2, CHAN3, CHAN4 or MATH.

Example :CURSor:MANual:SOURce CHANnel2 /*Set the channel source of the manual cursor measurement mode to CH2*/

:CURSor:MANual:TUNit

Syntax :CURSor:MANual:TUNit <unit>

:CURSor:MANual:TUNit?

Description Set or query the horizontal unit in the manual cursor measurement mode.

Parameter	Name	Type	Range	Default
	<unit>	Discrete	{S HZ DEGRee PERCent}	S

- Explanation**
- S: AX, BX and BX-AX in the measurement results are in "s" and 1/|dX| is in "Hz".
 - HZ: AX, BX and BX-AX in the measurement results are in "Hz" and 1/|dX| is in "s".
 - DEGRee: AX, BX and BX-AX are in "degree".
 - PERCent: AX, BX and BX-AX are in "%".

Return Format The query returns S, HZ, DEGR or PERC.

Example :CURSor:MANual:TUNit DEGRee /*Set the horizontal unit to "degree"*/

:CURSor:MANual:VUNit

Syntax :CURSor:MANual:VUNit <unit>

:CURSor:MANual:VUNit?

Description Set or query the vertical unit in the manual cursor measurement mode.

Parameter	Name	Type	Range	Default
	<unit>	Discrete	{PERCent SOURce}	SOURce

- Explanation**
- PERCent: AY, BY and BY-AY in the measurement results are in "%".
 - SOURce: the units of AY, BY and BY-AY in the measurement results will be automatically set to the unit of the current source.

Return Format The query returns PERC or SOUR.

Example :CURSor:MANual:VUNit PERCent /*Set the vertical unit to %*/

Related Command [:CHANnel<n>:UNITs](#)

:CURSor:MANual:AX

Syntax :CURSor:MANual:AX <x>

:CURSor:MANual:AX?

Description Set or query the horizontal position of cursor A in the manual cursor measurement mode.

Parameter	Name	Type	Range	Default
	<x>	Integer	5 to 594	100

Explanation The horizontal and vertical positions of the cursor are defined by the pixel coordinate of the screen. The pixel coordinate of the screen ranges from (0,0) to (600,400). Wherein, (0,0) is located at the left top corner of the screen and (600,400) is located at the right bottom corner of the screen. The horizontal pixel range is from 0 to 600 and the vertical pixel range is from 0 to 400.

Return The query returns an integer between 5 and 594.

Format

Example :CURSor:MANual:AX 200 /*Set the horizontal position of cursor A to 200*/

:CURSor:MANual:BX

Syntax :CURSor:MANual:BX <x>

:CURSor:MANual:BX?

Description Set or query the horizontal position of cursor B in the manual cursor measurement mode.

Parameter	Name	Type	Range	Default
	<x>	Integer	5 to 594	500

Explanation The horizontal and vertical positions of the cursor are defined by the pixel coordinate of the screen. The pixel coordinate of the screen ranges from (0,0) to (600,400). Wherein, (0,0) is located at the left top corner of the screen and (600,400) is located at the right bottom corner of the screen. The horizontal pixel range is from 0 to 600 and the vertical pixel range is from 0 to 400.

Return The query returns an integer between 5 and 594.

Format

Example :CURSor:MANual:BX 200 /*Set the horizontal position of cursor B to 200*/

:CURSor:MANual:AY

Syntax :CURSor:MANual:AY <y>

:CURSor:MANual:AY?

Description Set or query the vertical position of cursor A in the manual cursor measurement mode.

Parameter	Name	Type	Range	Default
	<y>	Integer	5 to 394	100

Explanation The horizontal and vertical positions of the cursor are defined by the pixel coordinate of the screen. The pixel coordinate of the screen ranges from (0,0) to (600,400). Wherein, (0,0) is located at the left top corner of the screen and (600,400) is located at the right bottom corner of the screen. The horizontal pixel range is from 0 to 600 and the vertical pixel range is from 0 to 400.

Return The query returns an integer between 5 and 394.

Format

Example :CURSor:MANual:AY 200 /*Set the vertical position of cursor A to 200*/

:CURSor:MANual:BY

Syntax :CURSor:MANual:BY <y>
:CURSor:MANual:BY?

Description Set or query the vertical position of cursor B in the manual cursor measurement mode.

Parameter	Name	Type	Range	Default
	<y>	Integer	5 to 394	300

Explanation The horizontal and vertical positions of the cursor are defined by the pixel coordinate of the screen. The pixel coordinate of the screen ranges from (0,0) to (600,400). Wherein, (0,0) is located at the left top corner of the screen and (600,400) is located at the right bottom corner of the screen. The horizontal pixel range is from 0 to 600 and the vertical pixel range is from 0 to 400.

Return Format The query returns an integer between 5 and 394.

Example :CURSor:MANual:BY 200 /*Set the vertical position of cursor B to 200*/

:CURSor:MANual:AXValue?

Syntax :CURSor:MANual:AXValue?

Description Query the X value of cursor A in the manual cursor measurement mode. The unit depends on the horizontal unit currently selected.

Return Format The query returns the X value of cursor A in scientific notation.

Example :CURSor:MANual:AXValue? /*The query returns -3.000000e-06*/

Related Commands [:CURSor:MANual:AX](#)
[:CURSor:MANual:TUNit](#)

:CURSor:MANual:AYValue?

Syntax :CURSor:MANual:AYValue?

Description Query the Y value of cursor A in the manual cursor measurement mode. The unit depends on the vertical unit currently selected.

Return Format The query returns the Y value of cursor A in scientific notation.

Example :CURSor:MANual:AYValue? /*The query returns -4.000000e-06*/

Related Commands [:CURSor:MANual:AY](#)
[:CURSor:MANual:VUNit](#)

:CURSor:MANual:BXValue?

Syntax :CURSor:MANual:BXValue?

Description Query the X value of cursor B in the manual cursor measurement mode. The unit depends on the horizontal unit currently selected.

Return Format The query returns the X value of cursor B in scientific notation.

Example :CURSor:MANual:BXValue? /*The query returns -3.000000e-06*/

Related Commands [:CURSor:MANual:BX](#)
[:CURSor:MANual:TUNit](#)

:CURSor:MANual:BYValue?

Syntax :CURSor:MANual:BYValue?

Description Query the Y value of cursor B in the manual cursor measurement mode. The unit depends on the vertical unit currently selected.

Return Format The query returns the Y value of cursor B in scientific notation.

Example :CURSor:MANual:BYValue? /*The query returns-4.000000e-06*/

Related Commands [:CURSor:MANual:BY](#)
[:CURSor:MANual:VUNit](#)

:CURSor:MANual:XDELta?

Syntax :CURSor:MANual:XDELta?

Description Query the difference between the X values of cursor A and cursor B (BX-AX) in the manual cursor measurement mode. The unit depends on the horizontal unit currently selected.

Return Format The query returns the difference in scientific notation.

Example :CURSor:MANual:XDELta? /*The query returns 6.120000e-06*/

Related Commands [:CURSor:MANual:AX](#)
[:CURSor:MANual:BX](#)
[:CURSor:MANual:TUNit](#)

:CURSor:MANual:IXDELta?

Syntax :CURSor:MANual:IXDELta?

Description Query the reciprocal of the absolute value of the difference between the X values of cursor A and cursor B ($1/|dX|$) in the manual cursor measurement mode. The unit depends on the horizontal unit currently selected.

Return Format The query returns $1/|dX|$ in scientific notation.

Example :CURSor:MANual:IXDELta? /*The query returns 1.120000e+05*/

Related Commands [:CURSor:MANual:AX](#)
[:CURSor:MANual:BX](#)
[:CURSor:MANual:TUNit](#)

:CURSor:MANual:YDELta?

Syntax :CURSor:MANual:YDELta?

Description Query the difference between the Y values of cursor A and cursor B (BY-AY) in the manual cursor measurement mode. The unit depends on the vertical unit currently selected.

Return Format The query returns the difference in scientific notation.

Example :CURSor:MANual:YDELta? /*The query returns -4.700000e+00*/

Related Commands [:CURSor:MANual:AY](#)
[:CURSor:MANual:BY](#)
[:CURSor:MANual:VUNit](#)

:CURSor:TRACk

Command List:

- ◆ [:CURSor:TRACk:SOURce1](#)
- ◆ [:CURSor:TRACk:SOURce2](#)
- ◆ [:CURSor:TRACk:AX](#)
- ◆ [:CURSor:TRACk:BX](#)
- ◆ [:CURSor:TRACk:AY?](#)
- ◆ [:CURSor:TRACk:BY?](#)
- ◆ [:CURSor:TRACk:AXValue?](#)
- ◆ [:CURSor:TRACk:AYValue?](#)
- ◆ [:CURSor:TRACk:BXValue?](#)
- ◆ [:CURSor:TRACk:BYValue?](#)
- ◆ [:CURSor:TRACk:XDELta?](#)
- ◆ [:CURSor:TRACk:IXDELTA?](#)

:CURSor:TRACk:SOURce1

Syntax :CURSor:TRACk:SOURce1 <source>
:CURSor:TRACk:SOURce1?

Description Set or query the channel source of cursor A in the track cursor measurement mode.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{OFF CHANnel1 CHANnel2 CHANnel3 CHANnel4 MATH}	CHANnel1

Explanation Only the channels enabled can be selected as the channel source.

Return Format The query returns OFF, CHAN1, CHAN2, CHAN3, CHAN4 or MATH.

Example :CURSor:TRACk:SOURce1 CHANnel2 /*Set the channel source to CH2*/

:CURSor:TRACk:SOURce2

Syntax :CURSor:TRACk:SOURce2 <source>
:CURSor:TRACk:SOURce2?

Description Set or query the channel source of cursor B in the track cursor measurement mode.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{OFF CHANnel1 CHANnel2 CHANnel3 CHANnel4 MATH}	CHANnel1

Explanation Only the channels enabled can be selected as the channel source.

Return Format The query returns OFF, CHAN1, CHAN2, CHAN3, CHAN4 or MATH.

Example :CURSor:TRACk:SOURce2 CHANnel2 /*Set the channel source to CH2*/

:CURSor:TRACk:AX

Syntax :CURSor:TRACk:AX <x>

:CURSor:TRACk:AX?

Description Set or query the horizontal position of cursor A in the track cursor measurement mode.

Parameter	Name	Type	Range	Default
	<x>	Integer	5 to 594	100

Explanation The horizontal and vertical positions of the cursor are defined by the pixel coordinate of the screen. The pixel coordinate of the screen ranges from (0,0) to (600,400). Wherein, (0,0) is located at the left top corner of the screen and (600,400) is located at the right bottom corner of the screen. The horizontal pixel range is from 0 to 600 and the vertical pixel range is from 0 to 400.

Return The query returns an integer between 5 and 594.

Format

Example :CURSor:TRACk:AX 200 /*Set the horizontal position of cursor A to 200*/

:CURSor:TRACk:BX

Syntax :CURSor:TRACk:BX <x>

:CURSor:TRACk:BX?

Description Set or query the horizontal position of cursor B in the track cursor measurement mode.

Parameter	Name	Type	Range	Default
	<x>	Integer	5 to 594	500

Explanation The horizontal and vertical positions of the cursor are defined by the pixel coordinate of the screen. The pixel coordinate of the screen ranges from (0,0) to (600,400). Wherein, (0,0) is located at the left top corner of the screen and (600,400) is located at the right bottom corner of the screen. The horizontal pixel range is from 0 to 600 and the vertical pixel range is from 0 to 400.

Return The query returns an integer between 5 and 594.

Format

Example :CURSor:TRACk:BX 200 /*Set the horizontal position of cursor B to 200*/

:CURSor:TRACk:AY?

Syntax :CURSor:TRACk:AY?

Description Query the vertical position of cursor A in the track cursor measurement mode.

Explanation

- The horizontal and vertical positions of the cursor are defined by the pixel coordinate of the screen. The pixel coordinate of the screen ranges from (0,0) to (600,400). Wherein, (0,0) is located at the left top corner of the screen and (600,400) is located at the right bottom corner of the screen. The horizontal pixel range is from 0 to 600 and the vertical pixel range is from 0 to 400.
- The setting is invalid when cursor A exceeds the vertical range of the screen display. At this point, 4294967295 will be returned.

Return The query returns an integer.

Format

Example :CURSor:TRACk:AY? /*The query returns 284*/

:CURSor:TRACk:BY?

Syntax :CURSor:TRACk:BY?

Description Query the vertical position of cursor B in the track cursor measurement mode.

- Explanation**
- The horizontal and vertical positions of the cursor are defined by the pixel coordinate of the screen. The pixel coordinate of the screen ranges from (0,0) to (600,400). Wherein, (0,0) is located at the left top corner of the screen and (600,400) is located at the right bottom corner of the screen. The horizontal pixel range is from 0 to 600 and the vertical pixel range is from 0 to 400.
 - The setting is invalid when cursor B exceeds the vertical range of the screen display. At this point, 4294967295 will be returned.

Return Format The query returns an integer.

Example :CURSor:TRACk:BY? /*The query returns 200*/

:CURSor:TRACk:AXValue?

Syntax :CURSor:TRACk:AXValue?

Description Query the X value of cursor A in the track cursor measurement mode. The default unit is s.

Return Format The query returns the X value of cursor A in scientific notation.

Example :CURSor:TRACk:AXValue? /*The query returns -3.000000e-06*/

Related Command [:CURSor:TRACk:AX](#)

:CURSor:TRACk:AYValue?

Syntax :CURSor:TRACk:AYValue?

Description Query the Y value of cursor A in the track cursor measurement mode. The unit is the same as the channel unit currently selected.

Return Format The query returns the Y value of cursor A in scientific notation.

Example :CURSor:TRACk:AYValue? /*The query returns -4.000000e-06*/

Related Commands [:CHANnel<n>:UNITs](#)
[:CURSor:TRACk:AY?](#)

:CURSor:TRACk:BXValue?

Syntax :CURSor:TRACk:BXValue?

Description Query the X value of cursor B in the track cursor measurement mode. The default unit is s.

Return Format The query returns the X value of cursor B in scientific notation.

Example :CURSor:TRACk:BXValue? /*The query returns -3.000000e-06*/

Related Command [:CURSor:TRACk:BX](#)

:CURSor:TRACk:BYValue?

Syntax :CURSor:TRACk:BYValue?

Description Query the Y value of cursor B in the track cursor measurement mode. The unit is the same as the channel unit currently selected.

Return Format The query returns the Y value of cursor B in scientific notation.

Example :CURSor:TRACk:BYValue? /*The query returns -4.000000e-06*/

Related Commands [:CHANnel<n>:UNITs](#)
[:CURSor:TRACk:BY?](#)

:CURSor:TRACk:XDELta?

Syntax :CURSor:TRACk:XDELta?

Description Query the difference between the X values of cursor A and cursor B (BX-AX) in the track cursor measurement mode. The default unit is s.

Return Format The query returns the difference in scientific notation.

Example :CURSor:TRACk:XDELta? /*The query returns 6.120000e-06*/

Related Commands [:CURSor:TRACk:AX](#)
[:CURSor:TRACk:BX](#)

:CURSor:TRACk:IXDELTA?

Syntax :CURSor:TRACk:IXDELTA?

Description Query the reciprocal of the absolute value of the difference between the X values of cursor A and cursor B ($1/|dX|$) in the track cursor measurement mode. The default unit is Hz.

Return Format The query returns $1/|dX|$ in scientific notation.

Example :CURSor:TRACk:IXDELTA? /*The query returns 1.120000e+05*/

Related Commands [:CURSor:TRACk:AX](#)
[:CURSor:TRACk:BX](#)

:CURSor:AUTO:ITEM

Syntax :COUSor:AUTO:ITEM <item>

:CURSor:AUTO:ITEM?

Description The auto cursor function can measure 24 waveform parameters. Using this command, you can select the parameters to be measured by the auto cursor from the five parameters enabled last or query the parameters currently measured by the auto cursor.

Parameter	Name	Type	Range	Default
	<item>	Discrete	{OFF ITEM1 ITEM2 ITEM3 ITEM4 ITEM5}	OFF

Explanation

- You can select the auto cursor measurement mode using the [:CURSor:MODE](#) command.
- The 24 waveform parameters are listed below (see the detailed introduction in [:MEASure Commands](#)). The parameters can be enabled by the [:MEASure:ITEM](#) command.
Period, Frequency, Rise Time, Fall Time, + Width, -Width, +Duty, -Duty, Delay 1→2, Delay 1→2, Phase 1→2, Phase 1→2, Vmax, Vmin, Vpp, Vtop, Vbase, Vamp, Vavg, Vrms, Overshoot, Preshoot, Area, Period Area.

Return Format The query returns OFF, ITEM1, ITEM2, ITEM3, ITEM4 or ITEM5.

Example :COUSor:AUTO:ITEM ITEM3

Related Commands [:CURSor:MODE](#)
[:MEASure:ITEM](#)

:CURSor:XY

The :CURSor:XY commands can only be used when the horizontal timebase mode is XY.

Command List:

- ◆ [:CURSor:XY:AX](#)
- ◆ [:CURSor:XY:BX](#)
- ◆ [:CURSor:XY:AY](#)
- ◆ [:CURSor:XY:BY](#)
- ◆ [:CURSor:XY:AXValue?](#)
- ◆ [:CURSor:XY:AYValue?](#)
- ◆ [:CURSor:XY:BXValue?](#)
- ◆ [:CURSor:XY:BYValue?](#)

:CURSor:XY:AX

Syntax :CURSor:XY:AX <x>
:CURSor:XY:AX?

Description Set or query the horizontal position of cursor A in the XY cursor measurement mode.

Parameter	Name	Type	Range	Default
	<x>	Integer	5 to 394	100

Explanation In the XY timebase mode, the horizontal and vertical positions of the cursor are defined by the pixel coordinate of the XY display area. The pixel coordinate of the screen ranges from (0,0) to (400,400). Wherein, (0,0) is located at the right top corner and (400,400) is located at the left bottom corner. The horizontal and vertical pixel ranges are both from 0 to 400.

Return Format The query returns an integer between 5 and 394.

Example :CURSor:XY:AX 200

:CURSor:XY:BX

Syntax :CURSor:XY:BX <x>
:CURSor:XY:BX?

Description Set or query the horizontal position of cursor B in the XY cursor measurement mode.

Parameter	Name	Type	Range	Default
	<x>	Integer	5 to 394	300

Explanation In the XY timebase mode, the horizontal and vertical positions of the cursor are defined by the pixel coordinate of the XY display area. The pixel coordinate of the screen ranges from (0,0) to (400,400). Wherein, (0,0) is located at the right top corner and (400,400) is located at the left bottom corner. The horizontal and vertical pixel ranges are both from 0 to 400.

Return Format The query returns an integer between 5 and 394.

Example :CURSor:XY:BX 200

:CURSor:XY:AY**Syntax** :CURSor:XY:AY <y>

:CURSor:XY:AY?

Description Set or query the vertical position of cursor A in the XY cursor measurement mode.

Parameter	Name	Type	Range	Default
	<x>	Integer	5 to 394	100

Explanation In the XY timebase mode, the horizontal and vertical positions are defined by the pixel coordinate of the XY display area. The pixel coordinate of the screen ranges from (0,0) to (400,400). Wherein, (0,0) is located at the right top corner and (400,400) is located at the left bottom corner. The horizontal and vertical pixel ranges are both from 0 to 400.**Return Format** The query returns an integer between 5 and 394.**Example** :CURSor:XY:AY 200**:CURSor:XY:BY****Syntax** :CURSor:XY:BY <y>

:CURSor:XY:BY?

Description Set or query the vertical position of cursor B in the XY cursor measurement mode.

Parameter	Name	Type	Range	Default
	<x>	Integer	5 to 394	300

Explanation In the XY timebase mode, the horizontal and vertical positions are defined by the pixel coordinate of the XY display area. The pixel coordinate of the XY display area ranges from (0,0) to (400,400). Wherein, (0,0) is located at the right top corner and (400,400) is located at the left bottom corner. The horizontal and vertical pixel ranges are both from 0 to 400.**Return Format** The query returns an integer between 5 and 394.**Example** :CURSor:XY:BY 200**:CURSor:XY:AXValue?****Syntax** :CURSor:XY:AXValue?**Description** Query the X value of cursor A in the XY cursor measurement mode. The unit depends on the amplitude unit of the corresponding channel.**Return Format** The query returns the X value of cursor A in scientific notation.**Example** :CURSor:XY:AXValue? /*The query returns 3.800000e-01*/**Related Command** [:CHANnel<n>:UNITs](#)

:CURSor:XY:AYValue?

Syntax :CURSor:XY:AYValue?

Description Query the Y value of cursor A in the XY cursor measurement mode. The unit depends on the amplitude unit of the corresponding channel.

Return Format The query returns the Y value of cursor A in scientific notation.

Example :CURSor:XY:AYValue? /*The query returns 4.000000e-01*/

Related Command [:CHANnel<n>:UNITs](#)

:CURSor:XY:BXValue?

Syntax :CURSor:XY:BXValue?

Description Query the X value of cursor B in the XY cursor measurement mode. The unit depends on the amplitude unit of the corresponding channel.

Return Format The query returns the X value of cursor B in scientific notation.

Example :CURSor:XY:BXValue? /*The query returns -4.200000e-01*/

Related Command [:CHANnel<n>:UNITs](#)

:CURSor:XY:BYValue?

Syntax :CURSor:XY:BYValue?

Description Query the Y value of cursor B in the XY cursor measurement mode. The unit depends on the amplitude unit of the corresponding channel.

Return Format The query returns the Y value of cursor B in scientific notation.

Example :CURSor:XY:BYValue? /*The query returns -4.000000e-01*/

Related Command [:CHANnel<n>:UNITs](#)

:DISPlay Commands

The :DISPlay commands can be used to set the waveform display mode, persistence time, waveform intensity, screen grid type and grid brightness.

Command List:

- ◆ [:DISPlay:CLEar](#)
- ◆ [:DISPlay:DATA?](#)
- ◆ [:DISPlay:TYPE](#)
- ◆ [:DISPlay:GRADing:TIME](#)
- ◆ [:DISPlay:WBRightness](#)
- ◆ [:DISPlay:GRID](#)
- ◆ [:DISPlay:GBRightness](#)

:DISPlay:CLEar

Syntax :DISPlay:CLEar

Description Clear all the waveforms on the screen.

- Explanation**
- If the oscilloscope is in the RUN state, waveform will still be displayed.
 - This command is equivalent to pressing the **CLEAR** key at the front panel. Sending the [:CLEar](#) command can also clear all the waveforms on the screen.

Related Commands [:RUN](#)
[:CLEar](#)

:DISPlay:TYPE**Syntax** :DISPlay:TYPE <type>

:DISPlay:TYPE?

Description Set or query the display mode of the waveform on the screen.

Parameter	Name	Type	Range	Default
	<type>	Discrete	{VECTors DOTS}	VECTors

Explanation ➤ VECTors: the sample points are connected by lines. Normally, this mode can provide the most vivid waveform to view the steep edge of the waveform (such as the square waveform).

➤ DOTS: display the sample points directly. You can directly view each sample point and use the cursor to measure the X and Y values of the sample point.

Return Format The query returns VECT or DOTS.**Example** :DISPlay:TYPE DOTS**:DISPlay:GRADing:TIME****Syntax** :DISPlay:GRADing:TIME <time>

:DISPlay:GRADing:TIME?

Description Set or query the persistence time. The default unit is s.

Parameter	Name	Type	Range	Default
	<time>	Discrete	{MIN 0.1 0.2 0.5 1 5 10 INFinite}	MIN

Explanation ➤ MIN: set the persistence time to its minimum to view the waveform changing in high refresh rate.

➤ Specific Values: set the persistence time to one of the values listed above to observe glitch that changes relatively slowly or glitch with low occurrence probability.

➤ INFinite: in this mode, the oscilloscope displays the newly acquired waveform without clearing the waveform formerly acquired. Enable to measure noise and jitter as well as capture incidental events.

Return Format The query returns the persistence time currently set.**Example** :DISPlay:GRADing:TIME 0.1 /*Set the persistence time to 0.1s*/**:DISPlay:WBRightness****Syntax** :DISPlay:WBRightness <time>

:DISPlay:WBRightness?

Description Set or query the waveform brightness. The default unit is %.

Parameter	Name	Type	Range	Default
	<time>	Integer	0 to 100	50

Return Format The query returns an integer between 0 and 100.**Example** :DISPlay:WBRightness 60

:DISPlay:GRID**Syntax** :DISPlay:GRID <grid>

:DISPlay:GRID?

Description Set or query the grid type of screen display.

Parameter	Name	Type	Range	Default
	<grid>	Discrete	{FULL HALF NONE}	FULL

Explanation FULL: turn the background grid and coordinate on.
 HALF: turn the background grid off.
 NONE: turn the background grid and coordinate off.

Return The query returns FULL, HALF or NONE.**Format****Example** :DISPlay:GRID NONE**:DISPlay:GBrightness****Syntax** :DISPlay:GBrightness <brightness>

:DISPlay:GBrightness?

Description Set or query the brightness of the screen grid. The default unit is %.

Parameter	Name	Type	Range	Default
	<brightness>	Integer	0 to 100	50

Return The query returns an integer between 0 and 100.**Format****Example** :DISPlay:GBrightness 60

IEEE488.2 Common Commands

The IEEE 488.2 standard defines some common commands used for querying the basic information of the instrument or executing the basic operations. These commands usually start with "*" and the keyword of the command is usually 3-character long.

Command List:

- ◆ [*CLS](#)
- ◆ [*ESE](#)
- ◆ [*ESR?](#)
- ◆ [*IDN?](#)
- ◆ [*OPC](#)
- ◆ [*RST](#)
- ◆ [*SRE](#)
- ◆ [*STB?](#)
- ◆ [*TST?](#)
- ◆ [*WAI](#)

*CLS

Syntax *CLS

Description Clear all the event registers in the register set and clear the error queue.

*ESE

Syntax *ESE <value>

*ESE?

Description Set or query the enable register for the standard event status register set.

Parameter	Name	Type	Range	Default
	<value>	Integer	0 to 255	0

Explanation The bit 1 and bit 6 of the standard event status register are not used and are always treated as 0, therefore, the range of <value> are the decimal numbers corresponding to the binary numbers ranging from 00000000 (0 in decimal) to 11111111 (255 in decimal) and of which the bit 1 and bit 6 are 0.

Return Format The query returns an integer which equals the sum of the weights of all the bits that have already been set in the register.

Example *ESE 16 /*Enable the bit 4 (16 in decimal) of the standard event status register */

*ESE? /*The query returns 16*/

ESR?*Syntax** *ESR?**Description** Query and clear the event register for the standard event status register.**Explanation** The bit 1 and bit 6 of the standard event status register are not used and are always treated as 0. The range of the return value are the decimal numbers corresponding to the binary numbers ranging from 00000000 (0 in decimal) to 11111111 (255 in decimal) and of which the bit 1 and bit 6 are 0.**Return Format** The query returns an integer between 0 and 255.***IDN?****Syntax** *IDN?**Description** Query the ID string of the instrument.**Return Format** The query returns the ID string of the instrument, such as RIGOL TECHNOLOGIES,DS1104Z,DS1T00000006,00.02.00.***OPC****Syntax** *OPC?

*OPC

Description The *OPC? command is used to query whether the current operation is finished. The *OPC command is used to set the Operation Complete bit (bit 0) in the standard event status register to 1 after the current operation is finished.**Return Format** The query returns 1 if the current operation is finished; otherwise, returns 0.***RST****Syntax** *RST**Description** Restore the instrument to the default state.***SRE****Syntax** *SRE <value>

*SRE?

Description Set or query the enable register for the status byte register set.**Parameter**

Name	Type	Range	Default
<value>	Integer	0 to 255	0

Explanation The bit 0 and bit 1 of the status byte register are not used and are always treated as 0, therefore, the range of <value> are the decimal numbers corresponding to the binary numbers ranging from 00000000 (0 in decimal) to 11111111 (255 in decimal) and of which the bit 0 and bit 1 are 0.**Return Format** The query returns an integer which equals the sum of the weights of all the bits that have already been set in the register.**Example** *SRE 16 /*Enable the bit 4 (16 in decimal) of the status byte register*/
*SRE? /*The query returns 16*/

***STB?**

Syntax *STB?

Description Query the event register for the status byte register. The value of the status byte register is set to 0 after this command is executed.

Explanation The bit 0 and bit 1 of the status byte register are not used and are always treated as 0. The query returns the decimal numbers corresponding to the binary numbers ranging from 00000000 (0 in decimal) to 11111111 (255 in decimal) and of which the bit 0 and bit 1 are 0.

Return Format The query returns an integer between 0 and 255.

***TST?**

Syntax *TST?

Description Perform a self-test and then returns the self-test results.

Return Format The query returns a decimal integer.

***WAI**

Syntax *WAI

Description Wait for the operation to finish.

Explanation The subsequent command can only be carried out after the current command has been executed.

:MATH Commands

The :MATH commands are used to set the operations between the waveforms of multiple channels.

Note:

- ✓ The operations can be divided into two types.
Analog Operations: $A+B$, $A-B$, $A \times B$, A/B , FFT, intg, diff, sqrt, lg, ln, exp and abs
Logic Operation: $A \& B$, $A || B$, A^B and $!A$
- ✓ For the logic operation, the waveform data to be operated is compared with the preset threshold and is converted to 0 or 1. Thus, the result will also be 0 or 1.
- ✓ For the analog operation or logic operation which needs only one source, only the [:MATH:SOURce1](#) command will be used.

Command List:

- ◆ [:MATH:DISPlay](#)
- ◆ [:MATH:OPERator](#)
- ◆ [:MATH:SOURce1](#)
- ◆ [:MATH:SOURce2](#)
- ◆ [:MATH:SCALe](#)
- ◆ [:MATH:OFFSet](#)
- ◆ [:MATH:INVert](#)
- ◆ [:MATH:RESet](#)
- ◆ [:MATH:FFT:WINDow](#)
- ◆ [:MATH:FFT:SPLit](#)
- ◆ [:MATH:FFT:UNIT](#)
- ◆ [:MATH:FFT:HSCale](#)
- ◆ [:MATH:FFT:HCENter](#)
- ◆ [:MATH:OPTion:STARt](#)
- ◆ [:MATH:OPTion:END](#)
- ◆ [:MATH:OPTion:INVert](#)
- ◆ [:MATH:OPTion:SENSitivity](#)
- ◆ [:MATH:OPTion:DIStance](#)
- ◆ [:MATH:OPTion:ASCale](#)
- ◆ [:MATH:OPTion:THReshold1](#)
- ◆ [:MATH:OPTion:THReshold2](#)

:MATH:DISPlay**Syntax** :MATH:DISPlay <bool>

:MATH:DISPlay?

Description Enable or disable the math operation function or query the math operation status.

Parameter	Name	Type	Range	Default
	<bool>	Bool	{{0 OFF} {1 ON}}	0 OFF

Return Format The query returns 0 or 1.**Example** :MATH:DISPlay ON**:MATH:OPERator****Syntax** :MATH:OPERator <opt>

:MATH:OPERator?

Description Set or query the operator of the math operation.

Parameter	Name	Type	Range	Default
	<opt>	Discrete	{ADD SUBTract MULTiply DIVision AND OR XOR NOT FFT INTG DIFF SQRT LOG LN EXP ABS}	ADD

Return Format The query returns ADD, SUBT, MULT, DIV, AND, OR, XOR, NOT, FFT, INTG, DIFF, SQRT, LOG, LN, EXP or ABS.**Example** :MATH:OPERator INTG**:MATH:SOURce1****Syntax** :MATH:SOURce1 <src>

:MATH:SOURce1?

Description Set or query the source of the math operation.

Parameter	Name	Type	Range	Default
	<src>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

- Explanation**
- For the operations which needs only one source, just this command will be used.
 - For the operations which needs two sources, this command is used to set source A.

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.**Example** :MATH:SOURce1 CHANnel3

:MATH:SOURce2

Syntax :MATH:SOURce2 <src>

:MATH:SOURce2?

Description Set or query the source of the math operation.

Parameter	Name	Type	Range	Default
	<src>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Explanation This command is only applicable to the operations which needs two source and it is used to set source B.

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :MATH:SOURce2 CHANnel3

:MATH:SCALE

Syntax :MATH:SCALE <scale>

:MATH:SCALE?

Description Set or query the vertical scale of the operation result. The unit depends on the operator currently selected and the unit of the source.

Parameter	Name	Type	Range	Default
	<scale>	Real	The max range is from 1p to 5T (in 1-2-5 step)	1.00V

Explanation The range of the vertical scale is related to the operator currently selected and the vertical scale of the source channel. For the integration (intg) and differential (diff) operations, it is also related to the current horizontal timebase.

Return Format The query returns the vertical scale of the operation result in scientific notation.

Example :MATH:SCALE 2 /*Set the vertical scale to 2*/
:MATH:SCALE? /*The query returns 2.000000e+00*/

Related Commands [:CHANnel<n>:SCALE](#)
[:CHANnel<n>:UNITs](#)
[:MATH:OPERator](#)
[:TIMebase\[:MAIN\]:SCALE](#)

:MATH:OFFSet

Syntax :MATH:OFFSet <offs>

:MATH:OFFSet?

Description Set or query the vertical offset of the operation result. The unit depends on the operator currently selected and the unit of the source.

Parameter	Name	Type	Range	Default
	<offs>	Real	Related to the vertical scale of the operation result. Range: $-1000 * \text{MathVerticalScale}$ to $1000 * \text{MathVerticalScale}$ Step: $\text{MathVerticalScale}/50$	0.00V

Explanation MathVerticalScale is the vertical scale of the operation result and can be set by the [:MATH:SCALE](#) command.

Return Format The query returns the vertical offset of the operation result in scientific notation.

Example :MATH:OFFSet 8 /*Set the vertical offset to 8V*/
:MATH:OFFSet? /*The query returns 8.000000e+00*/

Related Commands [:MATH:OPERator](#)
[:MATH:SCALE](#)
[:CHANnel<n>:UNITs](#)

:MATH:INVert

Syntax :MATH:INVert <bool>
:MATH:INVert?

Description Enable or disable the inverted display mode of the operation result, or query the inverted display mode status of the operation result.

Parameter	Name	Type	Range	Default
	<bool>	Bool	{{0 OFF}} {{1 ON}}	0 OFF

Explanation
➤ This command is invalid for the FFT operation.
➤ This command has the same function as the [:MATH:OPTion:INVert](#) command.

Return Format The query returns 0 or 1.

Example :MATH:INVert ON

Related Commands [:MATH:OPERator](#)
[:MATH:OPTion:INVert](#)

:MATH:RESet

Syntax :MATH:RESet

Description Sending this command, the instrument adjusts the vertical scale of the operation result to the most proper value according to the current operator and the horizontal timebase of the source.

Related Commands [:MATH:OPERator](#)
[:MATH:SCALE](#)
[:TIMEbase\[:MAIN\]:SCALE](#)

:MATH:FFT:WINDow

Syntax :MATH:FFT:WINDow <wnd>

:MATH:FFT:WINDow?

Description Set or query the window function of the FFT operation.

Parameter	Name	Type	Range	Default
<wnd>		Discrete	{RECTangle BLACkman HANNing HAMMing FLATop TRIangle}	RECTangle

- Explanation**
- Spectral leakage can be considerably decreased when a window function is used.
 - Different window functions are applicable to measure different waveforms. You need to select the window function according to waveform to be measured and its characteristics.

Return Format The query returns RECT, BLAC, HANN, HAMM, FLAT or TRI.

Example :MATH:FFT:WINDow BLACkman

Related Command [:MATH:OPERator](#)

:MATH:FFT:SPLit

Syntax :MATH:FFT:SPLit <bool>

:MATH:FFT:SPLit?

Description Enable or disable the half-screen display mode of the FFT operation, or query the status of the half display mode of the FFT operation.

Parameter	Name	Type	Range	Default
<bool>		Bool	{{0 OFF}} {1 ON}}	1 ON

- Explanation**
- Enable the half-screen display mode: the source channel and the FFT operation results are displayed separately. The time domain and frequency domain signals are displayed clearly.
 - Disable the half-screen display mode (full-screen display mode): the source channel and the FFT operation results are displayed in the same window to view the frequency spectrum more clearly and to perform more precise measurement.

Return Format The query returns 0 or 1.

Example :MATH:FFT:SPLit OFF

Related Command [:MATH:OPERator](#)

:MATH:FFT:UNIT

Syntax :MATH:FFT:UNIT <unit>

:MATH:FFT:UNIT?

Description Set or query the vertical unit of the FFT operation result.

Parameter	Name	Type	Range	Default
	<unit>	Discrete	{VRMS DB}	DB

Return Format The query returns DB or VRMS.

Example :MATH:FFT:UNIT VRMS

Related Command [:MATH:OPERator](#)

:MATH:FFT:HSCale

Syntax :MATH:FFT:HSCale <hsc>

:MATH:FFT:HSCale?

Description Set or query the horizontal scale of the FFT operation result. The default unit is Hz.

Parameter	Name	Type	Range	Default
	<hsc>	Real	Refer to Explanation	5.00MHz

- Explanation**
- <hsc> can be set to 1/20, 1/40, 1/100 and 1/200 of the current screen sample rate (current screen sample rate=100/current horizontal timebase).
For example, if the horizontal timebase of the instrument is 1 μ s, the current screen sample rate is 100/1 μ s=100MHz. Thus, the horizontal scale can be set to 5MHz, 2.5MHz, 1MHz or 500kHz.
 - You can view the detailed information of the spectrum by reducing the horizontal scale.

Return Format The query returns the horizontal scale in scientific notation.

Example :MATH:FFT:HSCale 500000 /*Set the horizontal scale to 500kHz*/
:MATH:FFT:HSCale? /*The query returns 5.000000e+05*/

Related Commands [:MATH:OPERator](#)
[:TIMEbase\[:MAIN\]:SCALE](#)

:MATH:FFT:HCEnter

Syntax :MATH:FFT:HCEnter <cent>

:MATH:FFT:HCEnter?

Description Set or query the center frequency of the FFT operation result, namely the frequency relative to the horizontal center of the screen. The default unit is Hz.

Parameter	Name	Type	Range	Default
	<cent>	Real	0 to (current screen sample rate×2/5)	5MHz

Explanation ➤ Current screen sample rate = 100/current horizontal timebase.

➤ Step = horizontal scale of the FFT operation result/50.

Return Format The query returns the current center frequency in scientific notation.

Example :MATH:FFT:HCEnter 10000000 /*Set the center frequency to 10MHz*/
:MATH:FFT:HCEnter? /*The query returns 1.000000e+07*/

Related Commands [:MATH:OPERator](#)
[:MATH:FFT:HSCale](#)
[:TIMebase\[:MAIN\]:SCALE](#)

:MATH:OPTion:STARt

Syntax :MATH:OPTion:STARt <sta>

:MATH:OPTion:STARt?

Description Set or query the start point of the waveform math operation.

Parameter	Name	Type	Range	Default
	<sta>	Integer	0 to (End point currently set-1)	0

Explanation ➤ This command is invalid for the FFT operation. Sending this command will modify the start points of all the operations (except FFT).

➤ The horizontal axis of the source selected is equally divided into 1200 parts, in which the leftmost is 0 and the rightmost is 1199.

Return Format The query returns an integer.

Example :MATH:OPTion:STARt 50

Related Command [:MATH:OPTion:END](#)

:MATH:OPTion:END

Syntax :MATH:OPTion:END <end>

:MATH:OPTion:END?

Description Set or query the end point of the waveform math operation.

Parameter	Name	Type	Range	Default
	<end>	Discrete	(Start point currently set+1) to 1199	1199

- Explanation**
- This command is invalid for the FFT operation. Sending this command will modify the end points of all the operations (except FFT).
 - The horizontal axis of the source selected is equally divided into 1200 parts, in which the leftmost is 0 and the rightmost is 1199.

Return Format The query returns an integer.

Example :MATH:OPTion:END 200

Related Command [:MATH:OPTion:START](#)

:MATH:OPTion:INVert

Syntax :MATH:OPTion:INVert <bool>

:MATH:OPTion:INVert?

Description Enable or disable the inverted display mode of the operation result, or query the inverted display mode status of the operation result.

Parameter	Name	Type	Range	Default
	<bool>	Bool	{{0 OFF}} {1 ON}}	0 OFF

- Explanation**
- This command is invalid for the FFT operation. Sending this command will modify the inverted display mode status of all the operations (except FFT).
 - This command has the same function as the [:MATH:INVert](#) command.

Return Format The query returns 0 or 1.

Example :MATH:OPTion:INVert ON

Related Command [:MATH:INVert](#)

:MATH:OPTion:SENSitivity

Syntax :MATH:OPTion:SENSitivity <sens>

:MATH:OPTion:SENSitivity?

Description Set or query the sensitivity of the logic operation. The default unit is div (namely the current vertical scale).

Parameter	Name	Type	Range	Default
	<sens>	Real	0 to 0.96div, the step is 0.08div	0

Explanation This command is only applicable to the logic operations (A&&B, A||B, A^B and !A).

Return Format The query returns the sensitivity in scientific notation.

Example :MATH:OPTion:SENSitivity 0.24 /*Set the sensitivity to 0.24div*/
:MATH:OPTion:SENSitivity? /*The query returns 2.400000e-01*/

Related Commands [:CHANnel<n>:SCALE](#)
[:MATH:OPERator](#)

:MATH:OPTion:DIStance

Syntax :MATH:OPTion:DIStance <dist>

:MATH:OPTion:DIStance?

Description Set or query the smoothing window width of the differential operation (diff).

Parameter	Name	Type	Range	Default
	<dist>	Integer	3 to 201	3

Explanation This command is only applicable to the differential operation (diff).

Return Format The query returns an integer between 3 and 201.

Example :MATH:OPTion:DIStance 25

Related Command [:MATH:OPERator](#)

:MATH:OPTion:ASCale

Syntax :MATH:OPTion:ASCale <opt>

:MATH:OPTion:ASCale?

Description Enable or disable the auto scale setting of the operation result or query the status of the auto scale setting.

Parameter	Name	Type	Range	Default
	<opt>	Bool	{{0 OFF} {1 ON}}	0 OFF

Explanation

- When the auto scale is enabled, the instrument will automatically calculate the vertical scale range according to the current operator, the vertical scale and the horizontal timebase. If the current scale is out of the range, it will adjust the vertical scale to the best value automatically.
- Sending this command will modify the auto scale status of all the operations.

Return Format The query returns 0 or 1.

Example :MATH:OPTion:ASCale ON

:MATH:OPTion:THReshold1

Syntax :MATH:OPTion:THReshold1 <thre>

:MATH:OPTion:THReshold1?

Description Set or query the threshold level of source A in the logic operation. The default unit is V.

Parameter	Name	Type	Range	Default
	<thre>	Real	(-4*Vertical Scale – Vertical Offset) to (4*Vertical Scale – Vertical Offset)	0.00V

- Explanation**
- This command is only applicable to the logic operations (A&&B, A||B, A^B and !A).
 - Vertical Scale refers to the vertical scale of source A
Vertical Offset refers to the vertical offset of source A
The step is Vertical Scale/25

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

Example :MATH:OPTion:THReshold1 0.8 /*Set the threshold level to 800mV*/
:MATH:OPTion:THReshold1? /*The query returns 8.000000e-01*/

Related Commands [:CHANnel<n>:SCALe](#)
[:CHANnel<n>:OFFSet](#)

:MATH:OPTion:THReshold2

Syntax :MATH:OPTion:THReshold2 <thre>

:MATH:OPTion:THReshold2?

Description Set or query the threshold level of source B in the logic operation. The default unit is V.

Parameter	Name	Type	Range	Default
	<thre>	Real	(-4*Vertical Scale–Vertical Offset) to (4*Vertical Scale–Vertical Offset)	0.00V

- Explanation**
- This command is only applicable to the logic operations (A&&B, A||B, A^B and !A).
 - Vertical Scale refers to the vertical scale of source B
Vertical Offset refers to the vertical offset of source B
The step is Vertical Scale/25

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

Example :MATH:OPTion:THReshold2 0.5

Related Commands [:CHANnel<n>:SCALe](#)
[:CHANnel<n>:OFFSet](#)

:MASK Commands

The :MASK commands are used to set and query the pass/fail test parameters.

Command List:

- ◆ [:MASK:ENABle](#)
- ◆ [:MASK:SOURce](#)
- ◆ [:MASK:OPERate](#)
- ◆ [:MASK:MDISplay](#)
- ◆ [:MASK:SOOutput](#)
- ◆ [:MASK:OUTPut](#)
- ◆ [:MASK:X](#)
- ◆ [:MASK:Y](#)
- ◆ [:MASK:CREate](#)
- ◆ [:MASK:PASSed?](#)
- ◆ [:MASK:FAILed?](#)
- ◆ [:MASK:TOTal?](#)
- ◆ [:MASK:RESet](#)

:MASK:ENABle

Syntax :MASK:ENABle <bool>

:MASK:ENABle?

Description Enable or disable the pass/fail test or query the status of the past/fail test.

Parameter	Name	Type	Range	Default
	<bool>	Bool	{{0 OFF}} {{1 ON}}	0 OFF

Explanation The pass/fail test is invalid in the following conditions: the horizontal timebase is XY or ROLL, in the slow sweep mode (namely when the horizontal timebase is YT, the horizontal timebase is set to 200ms/div or slower) as well as during waveform record.

Return Format The query returns 0 or 1.

Example :MASK:ENABle ON

:MASK:SOURce

Syntax :MASK:SOURce <source>

:MASK:SOURce?

Description Set or query the source of the pass/fail test.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Explanation Only the channel enabled can be selected as the source. You can sent the [:CHANnel<n>:DISPlay](#) command to enable the desired channel.

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :MASK:SOURce CHANnel2

Related Command [:CHANnel<n>:DISPlay](#)

:MASK:OPERate

Syntax :MASK:OPERate <oper>

:MASK:OPERate?

Description Run or stop the pass/fail test, or query the status of the pass/fail test.

Parameter	Name	Type	Range	Default
	<oper>	Discrete	{RUN STOP}	STOP

Explanation Before executing this command, you need to send the [:MASK:ENABle](#) command to enable the pass/fail test.

Return Format The query returns RUN or STOP.

Example :MASK:OPERate RUN

Related Command [:MASK:ENABle](#)

:MASK:MDISplay

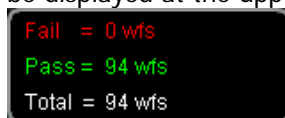
Syntax :MASK:MDISplay <bool>

:MASK:MDISplay?

Description Enable or disable the statistic information when the pass/fail test is enabled, or query the status of the statistic information.

Parameter	Name	Type	Range	Default
	<bool>	Bool	{{0 OFF}} {1 ON}}	0 OFF

- Explanation**
- Before executing this command, you need to send the [:MASK:ENABLE](#) command to enable the pass/fail test.
 - When the statistic information is on, the test results as shown in the figure below will be displayed at the upper right corner of the screen.



```
Fail = 0 wfs
Pass = 94 wfs
Total = 94 wfs
```

- You can send the [:MASK:PASSed?](#), [:MASK:FAILed?](#) and [:MASK:TOTal?](#) commands to query the test results.

Return Format The query returns 0 or 1.

Example :MASK:MDISplay ON

Related Commands

- [:MASK:ENABLE](#)
- [:MASK:PASSed?](#)
- [:MASK:FAILed?](#)
- [:MASK:TOTal?](#)

:MASK:SOOutput

Syntax :MASK:SOOutput <bool>

:MASK:SOOutput?

Description Turn the "Stop on Fail" function on or off, or query the status of the "Stop on Fail" function.

Parameter	Name	Type	Range	Default
	<bool>	Bool	{{0 OFF}} {1 ON}}	0 OFF

- Explanation**
- ON: when failed waveforms are detected, the oscilloscope will stop the test and enter the "STOP" state. At this point, the results of the test remain the same on the screen (if the display is turned on) and the **[Trigger Out]** connector (if enabled) at the rear panel outputs a single pulse.
 - OFF: the oscilloscope will continue with the test even though failed waveforms are detected. The test results on the screen will update continuously and the **[Trigger Out]** connector at the rear panel outputs a pulse each time a failed waveform is detected.

Return Format The query returns 0 or 1.

Example :MASK:SOOutput ON

:MASK:OUTPut**Syntax** :MASK:OUTPut <bool>

:MASK:OUTPut?

Description Enable or disable the sound prompt when the failed waveforms are detected, or query the status of the sound prompt.

Parameter	Name	Type	Range	Default
	<bool>	Bool	{{0 OFF}} {1 ON}}	0 OFF

- Explanation**
- OFF: when failed waveforms are detected, there are display and output but the beeper does not sound.
 - ON: when failed waveforms are detected, there are display and output and the beeper sounds (not related to the on/off state of the sound).

Return Format The query returns 0 or 1.**Example** :MASK:OUTPut ON**:MASK:X****Syntax** :MASK:X <x>

:MASK:X?

Description Set or query the horizontal adjustment parameter in the pass/fail test mask.

Parameter	Name	Type	Range	Default
	<x>	Real	0.02 to 4	0.24

Explanation The step is 0.02 within the range.**Return Format** The query returns the horizontal adjustment parameter in scientific notation.

Example :MASK:X 0.28 /*Set the horizontal adjustment parameter to 0.28div*/
 :MASK:X? /*The query returns 2.800000e-01*/

:MASK:Y**Syntax** :MASK:Y <y>

:MASK:Y?

Description Set or query the vertical adjustment parameter in the pass/fail test mask.

Parameter	Name	Type	Range	Default
	<y>	Real	0.04 to 5.12	0.48

Explanation The step is 0.04 within the range.**Return Format** The query returns the vertical adjustment parameter in scientific notation.

Example :MASK:Y 0.36 /*Set the vertical adjustment parameter to 0.36div*/
 :MASK:Y? /*The query returns 3.600000e-01*/

:MASK:CREate

Syntax :MASK:CREate

Description Create the pass/fail test mask using the current horizontal adjustment parameter and vertical adjustment parameter.

Explanation This command is valid only when the pass/fail test is enabled ([:MASK:ENABLE](#)) and is not in the run state ([:MASK:OPERate](#)).

**Related
Commands** [:MASK:ENABLE](#)
[:MASK:OPERate](#)
[:MASK:X](#)
[:MASK:Y](#)

:MASK:PASSed?

Syntax :MASK:PASSed?

Description Query the number of the passed frames in the pass/fail test.

**Return
Format** The query returns an integer.

:MASK:FAILed?

Syntax :MASK:FAILed?

Description Query the number of the failed frames in the pass/fail test.

**Return
Format** The query returns an integer.

:MASK:TOTal?

Syntax :MASK:TOTal?

Description Query the total number of the frames in the pass/fail test.

**Return
Format** The query returns an integer.

:MASK:RESet

Syntax :MASK:RESet

Description Reset the numbers of the passed frames and failed frames as well as the total number of frames in the pass/fail test to 0.

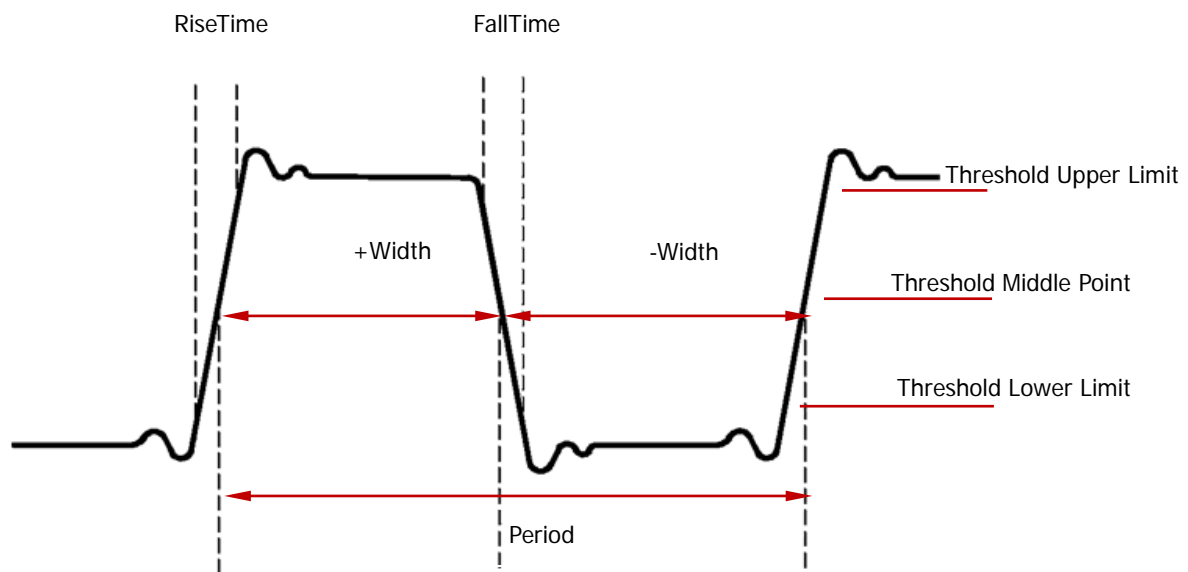
**Related
Commands** [:MASK:PASSed?](#)
[:MASK:FAILed?](#)
[:MASK:TOTal?](#)

:MEASure Commands

DS1000Z supports the auto measurement of the following 24 waveform parameters and provide the statistic function for the measurement results. In additional, you can use the frequency counter to make more precise frequency measurement. The :MEASure commands are used to set and query the measurement parameters.

➤ Time Parameters

- Period:** Defined as the time between the threshold middle points of two consecutive, like-polarity edges.
- Frequency:** Defined as the reciprocal of period.
- Rise Time:** The time for the signal amplitude to rise from the lower limit to the upper limit of the threshold.
- Fall Time:** The time for the signal amplitude to fall from the upper limit to the lower limit of the threshold.
- + Width:** The time difference between the threshold middle points of a rising edge to the next falling edge of the pulse.
- Width:** The time difference between the threshold middle points of a falling edge to the next rising edge of the pulse.
- + Duty:** The ratio of the positive pulse width to the period.
- Duty:** The ratio of the negative pulse width to the period.

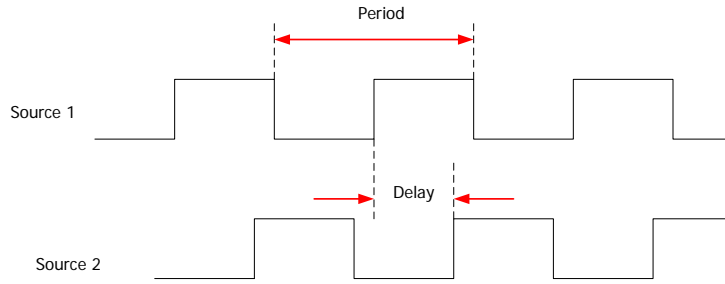


➤ Delay and Phase

- Delay 1→2 \uparrow :** The time difference between the rising edges of source 1 and source 2. Negative delay indicates that the selected rising edge of source 1 occurred after the selected rising edge of source 2.
- Delay 1→2 \downarrow :** The time difference between the falling edges of source 1 and source 2. Negative delay indicates that the selected falling edge of source 1 occurred after the selected falling edge of source 2.
- Phase 1→2 \uparrow :** Phase difference calculated according to "Delay 1→2 \uparrow " and the period of source 1, expressed in degree. The calculation formula is as shown below.
- Phase 1→2 \downarrow :** Phase difference calculated according to "Delay 1→2 \downarrow " and the period of source 1, expressed in degree. The calculation formula is as shown below.

$$Phase = \frac{Delay}{Period1} \times 360^\circ$$

Wherein, *Phase* denotes "Phase 1→2 $\frac{f}{T}$ " or "Phase 1→2 $\frac{T}{T}$ ", *Delay* denotes "Delay 1→2 $\frac{f}{T}$ " or "Delay 1→2 $\frac{T}{T}$ " and *Period1* denotes the period of source 1.



➤ **Voltage Parameters**

- Vmax:** the voltage value from the highest point of the waveform to the GND.
- Vmin:** the voltage value from the lowest point of the waveform to the GND.
- Vpp:** the voltage value from the highest point to the lowest point of the waveform.
- Vtop:** the voltage value from the flat top of the waveform to the GND.
- Vbase:** the voltage value from the flat base of the waveform to the GND.
- Vamp:** the voltage value from the top of the waveform to the base of the waveform.
- Vavg:** the arithmetic average value on the whole waveform or on the gating area.

$$Average = \frac{\sum_{i=1}^n x_i}{n}$$

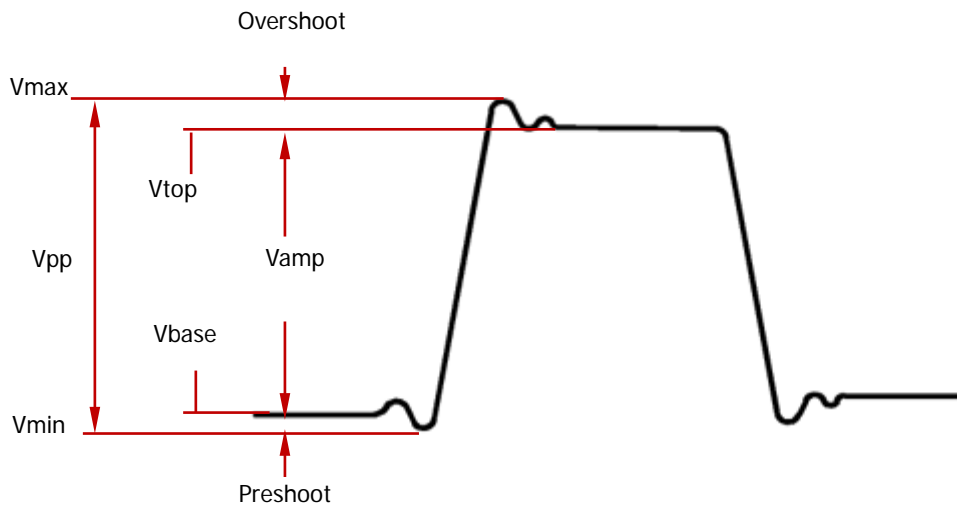
wherein, x_i is the *ith* point being measured, n is the number of points being measured.

- Vrms:** the root mean square value on the whole waveform or the gating area.

$$RMS = \sqrt{\frac{\sum_{i=1}^n x_i^2}{n}}$$

where, x_i is the *ith* point being measured, n is the number of points being measured.

- Overshoot:** the ratio of the difference between the maximum value and top value of the waveform to the amplitude value.
- Preshoot:** the ratio of the difference between the minimum value and base value of the waveform to the amplitude value.



➤ **Other Parameters**

Area 

The area of the whole waveform within the screen and the unit is voltage-second. The area measured above the zero reference (namely the vertical offset) is positive and the area measured below the zero reference is negative. The area measured is the algebraic sum of the area of the whole waveform within the screen.

Period Area 

The area of the first period of the waveform on the screen and the unit is voltage-second. The area above the zero reference (namely the vertical offset) is positive and the area below the zero reference is negative. The area measured is the algebraic sum of the area of the waveform within the whole period. Note that when the waveform on the screen is less than a period, the period area measured is 0.

Command List:

- ◆ [:MEASure:SOURce](#)
- ◆ [:MEASure:COUNter:SOURce](#)
- ◆ [:MEASure:COUNter:VALue?](#)
- ◆ [:MEASure:CLEar](#)
- ◆ [:MEASure:RECover](#)
- ◆ [:MEASure:ADISplay](#)
- ◆ [:MEASure:AMSource](#)
- ◆ [:MEASure:SETup:MAX](#)
- ◆ [:MEASure:SETup:MID](#)
- ◆ [:MEASure:SETup:MIN](#)
- ◆ [:MEASure:SETup:PSA](#)
- ◆ [:MEASure:SETup:PSB](#)
- ◆ [:MEASure:SETup:DSA](#)
- ◆ [:MEASure:SETup:DSB](#)
- ◆ [:MEASure:STATistic:DISPlay](#)
- ◆ [:MEASure:STATistic:MODE](#)
- ◆ [:MEASure:STATistic:RESet](#)
- ◆ [:MEASure:STATistic:ITEM](#)
- ◆ [:MEASure:ITEM](#)

:MEASure:SOURce

Syntax :MEASure:SOURce <sour>
:MEASure:SOURce?

Description Set or query the source of the current measurement parameter.

Parameter	Name	Type	Range	Default
	<sour>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Explanation This command is used to set the source for the measurement parameters (except the delay and phase). The delay and phase measurements requires two sources which can be set by the [:MEASure:SETup:DSA](#) and [:MEASure:SETup:DSB](#), [:MEASure:SETup:PSA](#) and [:MEASure:SETup:PSB](#) commands respectively.

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :MEASure:SOURce CHANnel2

Related Commands [:MEASure:SETup:PSA](#)
[:MEASure:SETup:PSB](#)
[:MEASure:SETup:DSA](#)
[:MEASure:SETup:DSB](#)
[:MEASure:ITEM](#)

:MEASure:COUNter:SOURce

Syntax :MEASure:COUNter:SOURce <sour>
:MEASure:COUNter:SOURce?

Description Set or query the source of the frequency counter, or disable the frequency counter.

Parameter	Name	Type	Range	Default
	<sour>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4 OFF}	OFF

Return Format The query returns CHAN1, CHAN2, CHAN3, CHAN4 or OFF.

Example :MEASure:COUNter:SOURce CHANnel2

Related Command [:MEASure:COUNter:VALue?](#)

:MEASure:COUNter:VALue?

Syntax :MEASure:COUNter:VALue?

Description Query the measurement result of the frequency counter. The default unit is Hz.

Return Format The query returns the measurement result in scientific notation. If the frequency counter is disabled, 0.000000e+00 will be returned.

Example :MEASure:COUNter:VALue? /*The query returns 1.000004e+03*/

Related Command [:MEASure:COUNter:SOURce](#)

:MEASure:CLEar

Syntax :MEASure:CLEar <item>

Description Clear one or all of the last five measurement items enabled.

Parameter	Name	Type	Range	Default
	<item>	Discrete	{ITEM1 ITEM2 ITEM3 ITEM4 ITEM5 ALL}	--

- Explanation**
- You can use the [:MEASure:ITEM](#) command to enable the desired parameters of the 24 waveform parameters. Note that the last five parameters are determined according to the order in which you turned them on and they will not change as you delete one or more measurement items.
 - You can send the [:MEASure:RECover](#) command to recover the item which has been cleared.

Example :MEASure:CLEar ITEM1

Related Commands [:MEASure:RECover](#)
[:MEASure:ITEM](#)

:MEASure:RECover

Syntax :MEASure:RECover <item>

Description Recover the measurement item which has been cleared.

Parameter	Name	Type	Range	Default
	<item>	Discrete	{ITEM1 ITEM2 ITEM3 ITEM4 ITEM5 ALL}	--

- Explanation**
- You can use the [:MEASure:ITEM](#) command to enable the desired parameters of the 24 waveform parameters. Note that the last five parameters are determined according to the order in which you turned them on and they will not change as you delete one or more measurement items.
 - You can send the [:MEASure:CLEar](#) command to clear the items again after one or all of the measurement items are recovered.

Related Commands [:MEASure:CLEar](#)
[:MEASure:ITEM](#)

:MEASure:ADISplay

Syntax :MEASure:ADISplay <bool>

:MEASure:ADISplay?

Description Enable or disable the all measurement function, or query the status of the all measurement function.

Parameter	Name	Type	Range	Default
	<bool>	Bool	{{0 OFF} {1 ON}}	0 OFF

Explanation

- The all measurement function can measure the following 20 parameters of the source at the same time:
Voltage Parameters: Vmax, Vmin, Vpp, Vtop, Vbase, Vamp, Vavg, Vrms, Overshoot and Peshoot
Time Parameters: Period, Frequency, Rise Time, Fall Time, + Width, - Width, + Duty and - Duty
Other Parameters: Area and Period Area.
- The all measurement function can measure CH1, CH2, CH3 and CH4 at the same time. You can send the [:MEASure:AMSource](#) command to set the source of the all measurement function.

Return Format The query returns 0 or 1.

Example :MEASure:ADISplay ON

Related Command [:MEASure:AMSource](#)

:MEASure:AMSource

Syntax :MEASure:AMSource <src>[,<src>[,<src>[,<src>]]]

:MEASure:AMSource?

Description Set or query the source of the all measurement function.

Parameter	Name	Type	Range	Default
	<src>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :MEASure:AMSource CHANnel1,CHANnel3

:MEASure:SETup:MAX

Syntax :MEASure:SETup:MAX <value>

:MEASure:SETup:MAX?

Description Set or query the upper limit of the threshold in the time, delay and phase measurements. The default unit is %.

Parameter	Name	Type	Range	Default
	<value>	Integer	7 to 95	90

- Explanation**
- The upper limit, middle point and lower limit of the threshold (expressed in %) are used to define the time, delay and phase parameters. Setting these values will affect the measurement results of the time, delay and phase parameters.
 - Setting the upper limit lower than the middle point will automatically reduce the middle point to keep it lower than the upper limit.

Return Format The query returns an integer between 7 and 95.

Example :MEASure:SETup:MAX 95

Related Commands [:MEASure:SETup:MID](#)
[:MEASure:SETup:MIN](#)
[:MEASure:ITEM](#)

:MEASure:SETup:MID

Syntax :MEASure:SETup:MID <value>

:MEASure:SETup:MID?

Description Set or query the middle point of the threshold in the time, delay and phase measurements. The default unit is %.

Parameter	Name	Type	Range	Default
	<value>	Integer	6 to 94	50

- Explanation**
- The upper limit, middle point and lower limit of the threshold (expressed in %) are used to define the time, delay and phase parameters. Setting these values will affect the measurement results of the time, delay and phase parameters.
 - The middle point must be lower than the upper limit and greater than the lower limit.

Return Format The query returns an integer between 6 and 94.

Example :MEASure:SETup:MID 89

Related Commands [:MEASure:SETup:MAX](#)
[:MEASure:SETup:MIN](#)
[:MEASure:ITEM](#)

:MEASure:SETup:MIN

Syntax :MEASure:SETup:MIN <value>

:MEASure:SETup:MIN?

Description Set or query the lower limit of the threshold in the time, delay and phase measurements. The default unit is %.

Parameter	Name	Type	Range	Default
	<value>	Integer	5 to 93	10

Explanation

- The upper limit, middle point and lower limit of the threshold (expressed in %) are used to define the time, delay and phase parameters. Setting these values will affect the measurement results of the time, delay and phase parameters.
- Setting the lower limit greater than the middle point will automatically increase the middle point to keep it greater than the upper limit.

Return Format The query returns an integer between 5 and 93.

Example :MEASure:SETup:MIN 93

Related Commands

- [:MEASure:SETup:MAX](#)
- [:MEASure:SETup:MID](#)
- [:MEASure:ITEM](#)

:MEASure:SETup:PSA

Syntax :MEASure:SETup:PSA <source>

:MEASure:SETup:PSA?

Description Set or query source A of the Phase 1→2_f and Phase 1→2_t measurements.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :MEASure:SETup:PSA CHANnel1

Related Command [:MEASure:ITEM](#)

:MEASure:SETup:PSB

Syntax :MEASure:SETup:PSB <source>

:MEASure:SETup:PSB?

Description Set or query source B of the Phase 1→2 $\frac{f}{T}$ and Phase 1→2 $\frac{T}{f}$ measurements.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel2

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :MEASure:SETup:PSB CHANnel1

Related Command [:MEASure:ITEM](#)

:MEASure:SETup:DSA

Syntax :MEASure:SETup:DSA <source>

:MEASure:SETup:DSA?

Description Set or query source A of the Delay 1→2 $\frac{f}{T}$ and Delay 1→2 $\frac{T}{f}$ measurements.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :MEASure:SETup:DSA CHANnel1

Related Command [:MEASure:ITEM](#)

:MEASure:SETup:DSB

Syntax :MEASure:SETup:DSB <source>

:MEASure:SETup:DSB?

Description Set or query source B of the Delay 1→2 $\frac{f}{T}$ and Delay 1→2 $\frac{T}{f}$ measurements.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel2

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :MEASure:SETup:DSB CHANnel1

Related Command [:MEASure:ITEM](#)

:MEASure:STATistic:DISPlay

Syntax :MEASure:STATistic:DISPlay <bool>

:MEASure:STATistic:DISPlay?

Description Enable or disable the statistic function, or query the status of the statistic function.

Parameter	Name	Type	Range	Default
	<bool>	Bool	{{0 OFF} {1 ON}}	0 OFF

Explanation When the statistic function is enabled, the oscilloscope makes statistic and displays the statistic results of at most 5 measurement items that are turned on last.

Return Format The query returns 0 or 1.

Example :MEASure:STATistic:DISPlay ON

Related Commands [:MEASure:STATistic:MODE](#)
[:MEASure:STATistic:RESet](#)
[:MEASure:STATistic:ITEM](#)
[:MEASure:ITEM](#)

:MEASure:STATistic:MODE

Syntax :MEASure:STATistic:MODE <mode>

:MEASure:STATistic:MODE?

Description Set or query the statistic mode.

Parameter	Name	Type	Range	Default
	<mode>	Discrete	{DIFFerence EXTRemum}	EXTRemum

Explanation

- DIFFerence: select the difference mode. The statistic results contain the current value, average value, standard deviation and counts.
- EXTRemum: select the extremum mode. The statistic results contain the current value, average value, minimum and maximum.
- Sending the [:MEASure:STATistic:DISPlay](#) command can enable the statistic function. When the statistic function is enabled, the oscilloscope makes statistic and displays the statistic results of at most 5 measurement items that are turned on last.

Return Format The query returns DIFF or EXTR.

Example :MEASure:STATistic:MODE DIFFerence /*Select the difference mode*/
:MEASure:STATistic:MODE? /*The query returns DIFF*/

Related Commands [:MEASure:STATistic:DISPlay](#)
[:MEASure:STATistic:RESet](#)
[:MEASure:STATistic:ITEM](#)

:MEASure:STATistic:RESet

Syntax :MEASure:STATistic:RESet

Description Clear the history data and make statistic again.

Related Commands [:MEASure:STATistic:DISPlay](#)
[:MEASure:STATistic:MODE](#)

[:MEASure:STATistic:ITEM](#)

:MEASure:STATistic:ITEM

Syntax :MEASure:STATistic:ITEM <item>[,<src>[,<src>]]
:MEASure:STATistic:ITEM? <type>,<item>[,<src>[,<src>]]

Description Enable the statistic function of any waveform parameter of the specified source, or query the statistic result of any waveform parameter of the specified source.

Parameter

Name	Type	Range	Default
<item>	Discrete	{VMAX VMIN VPP VTOP VBASe VAMP VAVG VRMS OVERshoot PREShoot MARea MPARea PERiod FREQuency RTIME FTIME PWIDth NWIDth PDUTy NDUTy RDELay FDELay RPHase FPHase}	--
<src>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	Refer to Explanation
<type>	Discrete	{MAXimum MINimum CURRent AVERages DEViation}	--

- Explanation**
- [,<src>[,<src>]] sets the source of the parameter to be measured.
 - If the parameter to be measured only needs a single source (such as VMAX, VMIN, VPP, VTOP, VBASe, VAMP, VAVG, VRMS, OVERshoot, MARea, MPARea, PREShoot, PERiod, FREQuency, RTIME, FTIME, PWIDth, NWIDth, PDUTy and NDUTy), you only need to set one source. If [,<src>[,<src>]] is omitted, the source is the one selected by the [:MEASure:SOURce](#) command by default.
 - If the parameter to be measured needs two sources (such as RDELay, FDELay, RPHase and FPHase), the command needs to include two sources; otherwise, the command is invalid. If [,<src>[,<src>]] is omitted, the sources are the ones selected by the [:MEASure:SETup:DSA](#) and [:MEASure:SETup:DSB](#) or [:MEASure:SETup:PSA](#) and [:MEASure:SETup:PSB](#) commands by default.

Return Format The query returns the statistic result in scientific notation.

Example :MEASure:STATistic:ITEM VPP,CHANnel2 /*Enable the Vpp statistic function of CH2*/
:MEASure:STATistic:ITEM? MAXimum,VPP /*The query returns 9.120000e-01*/

Related Commands [:MEASure:SOURce](#)
[:MEASure:SETup:DSA](#)
[:MEASure:SETup:DSB](#)
[:MEASure:SETup:PSA](#)
[:MEASure:SETup:PSB](#)

:MEASure:ITEM

Syntax :MEASure:ITEM <item>[,<src>[,<src>]]

:MEASure:ITEM? <item>[,<src>[,<src>]]

Description Measure any waveform parameter of the specified source, or query the measurement result of any waveform parameter of the specified source.

Parameter	Name	Type	Range	Default
<item>		Discrete	{VMAX VMIN VPP VTOP VBASE VAMP VAVG VRMS OVERshoot PREShoot MAREa MPAREa PERiod FREQuency RTIME FTIME PWIDth NWIDth PDUTy NDUTy RDElay FDElay RPHase FPHase }	--
<src>		Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	--

- Explanation**
- [,<src>[,<src>]] sets the source of the parameter to be measured.
 - If the parameter to be measured only needs a single source (such as VMAX, VMIN, VPP, VTOP, VBASE, VAMP, VAVG, VRMS, OVERshoot, MAREa, MPAREa, PREShoot, PERiod, FREQuency, RTIME, FTIME, PWIDth, NWIDth, PDUTy and NDUTy), you only need to set one source. If [,<src>[,<src>]] is omitted, the source is the one selected by the [:MEASure:SOURce](#) command by default.
 - If the parameter to be measured needs two sources (such as RDElay, FDElay, RPHase and FPHase), the command needs to include two sources; otherwise, the command is invalid. If [,<src>[,<src>]] is omitted, the sources are the ones selected by the [:MEASure:SETup:DSA](#) and [:MEASure:SETup:DSB](#) or [:MEASure:SETup:PSA](#) and [:MEASure:SETup:PSB](#) commands by default.

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

Example :MEASure:ITEM OVERshoot,CHANnel2 /*Enable the overshoot measurement of CH2*/
:MEASure:ITEM? OVERshoot,CHANnel2 /*The query returns 8.888889e-03*/

Related Commands

- [:MEASure:SOURce](#)
- [:MEASure:SETup:DSA](#)
- [:MEASure:SETup:DSB](#)
- [:MEASure:SETup:PSA](#)
- [:MEASure:SETup:PSB](#)

:REference Commands

Command List:

- ◆ [:REference:DISPlay](#)
- ◆ [:REference<n>:ENABle](#)
- ◆ [:REference<n>:SOURce](#)
- ◆ [:REference<n>:VSCale](#)
- ◆ [:REference<n>:VOFFset](#)
- ◆ [:REference<n>:RESet](#)

:REference:DISPlay

Syntax :REference:DISPlay <onoff>
:REference:DISPlay?

Description Enable or disable the REF function, or query the status of the REF function.

Parameter	Name	Type	Range	Default
	<onoff>	Bool	{{0 OFF} {1 ON}}	0 OFF

Return Format The query returns 0 or 1.

Example :REference:DISPlay ON

:REference<n>:ENABle

Syntax :REference<n>:ENABle <bool>
:REference<n>:ENABle?

Description Enable or disable the specified reference channel, or query the status of the specified reference channel.

Parameter	Name	Type	Range	Default
	<n>	Discrete	{1 2 3 4 5 6 7 8 9 10}	1
	<bool>	Bool	{{0 OFF} {1 ON}}	Ref1: 1 ON Others: 0 OFF

Return Format The query returns 0 or 1.

Example :REference1:ENABle ON

:REfERENCE<n>:SOURce

Syntax :REfERENCE<n>:SOURce <source>

:REfERENCE<n>:SOURce?

Description Set or query the source of the specified reference channel.

Parameter	Name	Type	Range	Default
	<n>	Discrete	{1 2 3 4 5 6 7 8 9 10}	None
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4 MATH}	CHANnel1

Explanation Among CH1, CH2, CH3, CH4 and MATH, only the channel currently enabled can be selected as the source or the specified reference channel.

Return Format The query returns CHAN1, CHAN2, CHAN3, CHAN4 or MATH.

Example :REfERENCE1:SOURce CHANnel1

:REfERENCE<n>:VSCale

Syntax :REfERENCE<n>:VSCale <scale>

:REfERENCE<n>:VSCale?

Description Set or query the vertical scale of the specified reference channel. The unit is the same as the unit of the source.

Parameter	Name	Type	Range	Default
	<n>	Discrete	{1 2 3 4 5 6 7 8 9 10}	None
	<scale>	Real	Related to the probe ratio: when probe ratio is 1: 1mV to 10V	The vertical scale of the source when saving the reference waveform

Explanation This command is valid only when the specified reference channel has stored reference waveform. Otherwise, the oscilloscope will display "No reference data!" when sending this command.

Return Format The query returns the vertical scale in scientific notation.

Example :REfERENCE1:VSCale 2

Related Command [:REfERENCE<n>:SOURce](#)

:REFerence<n>:VOFFset

Syntax :REFerence<n>:VOFFset <offset>

:REFerence<n>:VOFFset?

Description Set or query the vertical offset of the specified reference channel. The unit is the same as the unit of the source.

Parameter	Name	Type	Range	Default
	<n>	Discrete	{1 2 3 4 5 6 7 8 9 10}	None
	<offset>	Real	-10*RefVerticalScale to 10*RefVerticalScale	0

Explanation RefVerticalScale refers to the vertical scale of the reference channel.

Return Format The query returns the vertical offset in scientific notation.

Example :REFerence1:VOFFset 10

Related Commands [:REFerence<n>:SOURce](#)
[:REFerence<n>:VSCale](#)

:REFerence<n>:RESet

Syntax :REFerence<n>:RESet

Description Reset the vertical scale and vertical offset of the specified reference channel to their default values.

Parameter	Name	Type	Range	Default
	<n>	Discrete	{1 2 3 4 5 6 7 8 9 10}	None

[:SOURce[<n>]] Commands (DS1000Z-S Only)

Command List:

- ◆ [\[:SOURce\[<n>\]\]:OUTPut\[<n>\]\[:STATe\]](#)
- ◆ [\[:SOURce\[<n>\]\]:OUTPut\[<n>\]:IMPedance](#)
- ◆ [\[:SOURce\[<n>\]\]:FREQuency\[:FIXed\]](#)
- ◆ [\[:SOURce\[<n>\]\]:PHASe\[:ADJust\]](#)
- ◆ [\[:SOURce\[<n>\]\]:PHASe:INITiate](#)
- ◆ [\[:SOURce\[<n>\]\]:FUNCTion\[:SHAPE\]](#)
- ◆ [\[:SOURce\[<n>\]\]:FUNCTion:RAMP:SYMMetry](#)
- ◆ [\[:SOURce\[<n>\]\]:VOLTage\[:LEVel\]\[:IMMEDIATE\]\[:AMPLitude\]](#)
- ◆ [\[:SOURce\[<n>\]\]:VOLTage\[:LEVel\]\[:IMMEDIATE\]:OFFSet](#)
- ◆ [\[:SOURce\[<n>\]\]:PULSe:DCYCLE](#)
- ◆ [\[:SOURce\[<n>\]\]:MOD\[:STATe\]](#)
- ◆ [\[:SOURce\[<n>\]\]:MOD:TYPE](#)
- ◆ [\[:SOURce\[<n>\]\]:MOD:AM\[:DEPTH\]](#)
- ◆ [\[:SOURce\[<n>\]\]:MOD:AM:INTernal:FREQuency](#)
- ◆ [\[:SOURce\[<n>\]\]:MOD:FM:INTernal:FREQuency](#)
- ◆ [\[:SOURce\[<n>\]\]:MOD:AM:INTernal:FUNCTion](#)
- ◆ [\[:SOURce\[<n>\]\]:MOD:FM:INTernal:FUNCTion](#)
- ◆ [\[:SOURce\[<n>\]\]:MOD:FM\[:DEVlation\]](#)
- ◆ [\[:SOURce\[<n>\]\]:APPLY?](#)
- ◆ [\[:SOURce\[<n>\]\]:APPLY:NOISe](#)
- ◆ [\[:SOURce\[<n>\]\]:APPLY:PULSe](#)
- ◆ [\[:SOURce\[<n>\]\]:APPLY:RAMP](#)
- ◆ [\[:SOURce\[<n>\]\]:APPLY:SINusoid](#)
- ◆ [\[:SOURce\[<n>\]\]:APPLY:SQUare](#)
- ◆ [\[:SOURce\[<n>\]\]:APPLY:USER](#)

[[:SOURce[<n>]]:OUTPut[<n>][:STATe]

Syntax [:SOURce[<n>]]:OUTPut[<n>][:STATe] <bool>
[:SOURce[<n>]]:OUTPut[<n>][:STATe]?

Description Turn on or off the output of the specified source channel, or query the status of the output of the specified source channel.

Parameter	Name	Type	Range	Default
	[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
	<bool>	Bool	{{0 OFF} {1 ON}}	0 OFF

Explanation [:SOURce[<n>]] and [<n>] are used to specify the source channel. The former enjoys higher priority over the latter, namely if [:SOURce[<n>]] is omitted, the channel is specified by [<n>] and if [:SOURce[<n>]] is not omitted, the channel is specified by [:SOURce[<n>]]. If both are omitted, the operation is performed on source 1 by default.

Return Format The query returns OFF or ON.

Example :OUTPut 1

[[:SOURce[<n>]]:OUTPut[<n>]:IMPedance

Syntax [:SOURce[<n>]]:OUTPut[<n>]:IMPedance <impedance>
[:SOURce[<n>]]:OUTPut[<n>]:IMPedance?

Description Set or query the impedance of the specified source channel.

Parameter	Name	Type	Range	Default
	[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
	<impedance>	Discrete	{OMEG FIFTy}	OMEG

Explanation

- OMEG: high impedance; FIFTy: 50Ω
- [:SOURce[<n>]] and [<n>] are used to specify the source channel. The former enjoys higher priority over the latter, namely if [:SOURce[<n>]] is omitted, the channel is specified by [<n>] and if [:SOURce[<n>]] is not omitted, the channel is specified by [:SOURce[<n>]]. If both are omitted, the operation is performed on source 1 by default.

Return Format The query returns OMEG or FIFT.

Example :OUTPut:IMPedance FIFTy

[:SOURce[<n>]]:FREQuency[:FIXed]

Syntax [:SOURce[<n>]]:FREQuency[:FIXed] <frequency>
[:SOURce[<n>]]:FREQuency[:FIXed]?

Description Set or query the output frequency of the specified source channel if the modulation is not enabled or the carrier frequency if the modulation is enabled. The default unit is Hz.

Parameter	Name	Type	Range	Default
	[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
	<frequency>	Real	Sine: 0.1Hz to 25MHz Square: 0.1Hz to 15MHz Pulse: 0.1Hz to 1MHz Ramp: 0.1Hz to 100kHz Arbitrary waveform: 0.1Hz to 10MHz	100kHz

Return Format The query returns the frequency in scientific notation, such as 2.0000000e+05.

Example :FREQuency 1000

Related Commands [\[:SOURce\[<n>\]:MOD:STATe\]](#)
[\[:SOURce\[<n>\]:MOD:TYPe\]](#)

[:SOURce[<n>]]:PHASe[:ADJust]

Syntax [:SOURce[<n>]]:PHASe[:ADJust] <phase>
[:SOURce[<n>]]:PHASe[:ADJust]?

Description Set or query the start phase of the specified source channel. The default unit is degree (°).

Parameter	Name	Type	Range	Default
	[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
	<phase>	Real	0 to 360	0

Return Format The query returns the start phase in scientific notation, such as 0.0000000e+00.

Example :PHASe 90

[[:SOURce[<n>]]:PHASe:INITiate

Syntax [[:SOURce[<n>]]:PHASe:INITiate

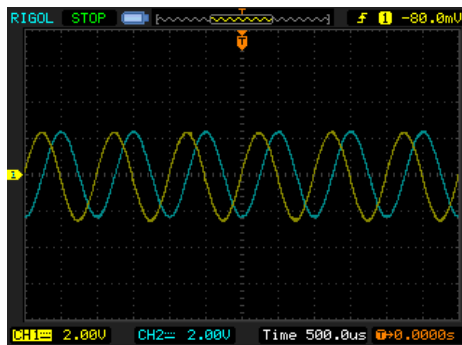
Description Execute the align phase operation.

Parameter

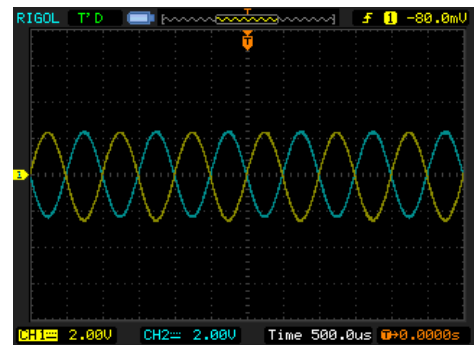
Name	Type	Range	Default
[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default

Explanation When the align phase operation is performed, the instrument re-configures the two channels to make them output with the specified frequency and phase.

For two signals whose frequencies are the same or in multiple, this operation will align their phases. For example, assume Source1 output a 1kHz, 5Vpp, 0° sine waveform and Source2 outputs a 1kHz, 5Vpp, 180° sine waveform. Sample the two waveforms using another oscilloscope and display them stably; you will see that the waveforms shown on the oscilloscope do not always have a phase deviation of 180°. At this point, execute the align phase operation and the waveforms shown on the oscilloscope will have a phase deviation of 180° without any manual adjustment of the start phase of the generator.



Before Aligning Phase



After Aligning Phase

[[:SOURce[<n>]]:FUNction[:SHAPe]

Syntax [[:SOURce[<n>]]:FUNction[:SHAPe] <wave>

[[:SOURce[<n>]]:FUNction[:SHAPe]?

Description Select or query the output waveform when the modulation of the specified source channel is not enabled. Select or query the carrier waveform when the modulation is enabled. At this point, if PULSe, NOISe or DC is selected, the modulation will turn off automatically.

Parameter

Name	Type	Range	Default
[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
<wave>	Discrete	{SINusoid SQUare RAMP PULSe NOISe DC INTErnal EXTernal}	SINusoid

Explanation

- INTERNAL: built-in waveforms (7 built-in waveforms are provided by DS1000Z, including Sinc, Exp.Rise, Exp.Fall, ECG, Gauss, Lorentz and Haversine)
- EXTERNAL: arbitrary waveform

Return Format The query returns SIN, SQU, RAMP, PULS, NOIS, DC, INTE or EXT.

Example :FUNction SQUare

[:SOURce[<n>]]:FUNction:RAMP:SYMMetry

Syntax [:SOURce[<n>]]:FUNction:RAMP:SYMMetry <val>

[:SOURce[<n>]]:FUNction:RAMP:SYMMetry?

Description Set or query the ramp symmetry (the percentage that the rising period takes up in the whole period) of the specified source channel. The default unit is %.

Parameter

Name	Type	Range	Default
[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
<val>	Real	0 to 100	10

Return Format The query returns the symmetry in scientific notation, such as 5.000000e+01.

Example :FUNction:RAMP:SYMMetry 50

Related Command [\[:SOURce\[<n>\]\]:FUNction\[:SHAPE\]](#)

[:SOURce[<n>]]:VOLTage[:LEVel][:IMMediate][:AMPLitude]

Syntax [:SOURce[<n>]]:VOLTage[:LEVel][:IMMediate][:AMPLitude] <Amplitude>

[:SOURce[<n>]]:VOLTage[:LEVel][:IMMediate][:AMPLitude]?

Description Set or query the output amplitude of the specified source channel. The default unit is Vpp.

Parameter

Name	Type	Range	Default
[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
<Amplitude>	Real	Related to the output impedance: HighZ: 20mVpp to 5Vpp 50Ω: 10mVpp to 2.5Vpp	1Vpp

Explanation You can use the [\[:SOURce\[<n>\]\]:OUTPut\[<n>\]:IMPedance](#) command to set the output impedance.

Return Format The query returns the amplitude in scientific notation, such as 1.0000000e+00.

Example :VOLTage 2

[:SOURce[<n>]]:VOLTage[:LEVel][:IMMediate]:OFFSet

Syntax [:SOURce[<n>]]:VOLTage[:LEVel] [:IMMediate]:OFFSet <offset>
[:SOURce[<n>]]:VOLTage[:LEVel] [:IMMediate]:OFFSet?

Description Set or query the DC offset of the specified source channel. The default unit is V.

Parameter	Name	Type	Range	Default
	[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
	<offset>	Real	Related to the output impedance and amplitude: HighZ: $\pm(2.5V\text{-current amplitude}/2)$ 50 Ω : $\pm(1.25V\text{-current amplitude}/2)$	0V

- Explanation**
- You can use the [\[:SOURce\[<n>\]\]:OUTPut\[<n>\]:IMPedance](#) command to set the output impedance.
 - You can use the [\[:SOURce\[<n>\]\]:VOLTage\[:LEVel\]\[:IMMediate\]\[:AMPLitude\]](#) command to set the output amplitude.

Return Format The query returns the DC offset in scientific notation, such as 1.0000000e+00.

Example :VOLTage:OFFSet 0.5

[:SOURce[<n>]]:PULSe:DCYClE

Syntax [:SOURce[<n>]]:PULSe:DCYClE <percent>
[:SOURce[<n>]]:PULSe:DCYClE?

Description Set or query the pulse duty cycle (the percentage that the high level takes up in the whole period) of the specified source channel. The default unit is %.

Parameter	Name	Type	Range	Default
	[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
	<percent>	Real	10 to 90	20

Return Format The query returns the duty cycle in scientific notation, such as 5.000000e+01.

Example :PULSe:DCYClE 50

Related Command [\[:SOURce\[<n>\]\]:FUNCTion\[:SHAPE\]](#)

[**:SOURce**[<n>]]:**MOD**[:**STATe**]

Syntax [**:SOURce**[<n>]]:**MOD**[:**STATe**] <bool>

[**:SOURce**[<n>]]:**MOD**[:**STATe**]?

Description Enable or disable the modulation of the specified source channel, or query the status of the modulation of the specified source channel.

Parameter	Name	Type	Range	Default
	[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
	<bool>	Bool	{{1 ON} {0 OFF}}	0 OFF

- Explanation**
- The signal source of DS1000Z provides two types of modulations (Amplitude Modulation (AM) and Frequency Modulation (FM)) which can be selected by sending the [\[:SOURce\[<n>\]\]:MOD:TYPE](#) command.
 - Sine, Square, Ramp, Built-in waveforms or Arb waveform (except DC) can be used as the carrier waveform.
 - You can use the [\[:SOURce\[<n>\]\]:MOD:AM:INTernal:FUNCTION](#) or [\[:SOURce\[<n>\]\]:MOD:FM:INTernal:FUNCTION](#) command to select Sine, Square, Ramp or Noise as the modulating waveform.

Return Format The query returns OFF or ON.

Example :MOD ON

Related Commands [\[:SOURce\[<n>\]\]:FUNCTION\[:SHAPE\]](#)

[\[:SOURce\[<n>\]\]:MOD:TYPE](#)

[\[:SOURce\[<n>\]\]:MOD:AM:INTernal:FUNCTION](#)

[\[:SOURce\[<n>\]\]:MOD:FM:INTernal:FUNCTION](#)

[**:SOURce**[<n>]]:**MOD**:**TYPE**

Syntax [**:SOURce**[<n>]]:**MOD**:**TYPE** <type>

[**:SOURce**[<n>]]:**MOD**:**TYPE**?

Description Set or query the modulation type of the specified source channel.

Parameter	Name	Type	Range	Default
	[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
	<type>	Discrete	{AM FM}	AM

- Explanation**
- AM: amplitude modulation. The amplitude of the carrier waveform varies with the amplitude of the modulating waveform.
 - FM: frequency modulation. The frequency of the carrier waveform varies with the amplitude of the modulating waveform.
 - Sine, Square, Ramp, Built-in waveforms or Arb waveform (except DC) can be used as the carrier waveform.
 - You can use the [\[:SOURce\[<n>\]\]:MOD:AM:INTernal:FUNCTION](#) or [\[:SOURce\[<n>\]\]:MOD:FM:INTernal:FUNCTION](#) command to select Sine, Square, Ramp or Noise as the modulating waveform.

Return Format The query returns AM or FM.

Example :MOD:TYPE AM

[[:SOURce[<n>]]:MOD:AM[:DEPTh]

Syntax [:SOURce[<n>]]:MOD:AM[:DEPTh] <depth>
[:SOURce[<n>]]:MOD:AM[:DEPTh]?

Description Set or query the AM modulation depth (indicates the amplitude variation degree and is expressed as a percentage) of the specified source channel. The default unit is %.

Parameter	Name	Type	Range	Default
	[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
	<depth>	Real	0 to 120	100

Explanation In 0% modulation, the output amplitude is half of the carrier waveform amplitude. In 100% modulation, the output amplitude is equal to the carrier waveform amplitude. In >100% modulation, envelop distortion which should be avoided in the real circuit will occur and the output amplitude of the instrument would not exceed 5Vpp (50Ω load).

Return Format The query returns an integer.

Example :MOD:AM 80

**[[:SOURce[<n>]]:MOD:AM:INTernal:FREQuency
[:SOURce[<n>]]:MOD:FM:INTernal:FREQuency**

Syntax [:SOURce[<n>]]:MOD:AM:INTernal:FREQuency <frequency>
[:SOURce[<n>]]:MOD:AM:INTernal:FREQuency?
[:SOURce[<n>]]:MOD:FM:INTernal:FREQuency <frequency>
[:SOURce[<n>]]:MOD:FM:INTernal:FREQuency?

Description Set or query the modulating waveform frequency of AM or FM of the specified source channel. The default unit is Hz.

Parameter	Name	Type	Range	Default
	[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
	<frequency>	Real	1Hz to 50kHz	1kHz

Explanation

- Sending the [\[:SOURce\[<n>\]\]:MOD:TYPE](#) command can set the modulation type.
- AM: amplitude modulation. The amplitude of the carrier waveform varies with the amplitude of the modulating waveform.
FM: frequency modulation. The frequency of the carrier waveform varies with the amplitude of the modulating waveform.
- Sending [\[:SOURce\[<n>\]\]:MOD:AM:INTernal:FUNCTION](#) or [\[:SOURce\[<n>\]\]:MOD:FM:INTernal:FUNCTION](#) can select Sine, Square, Ramp or Noise as the modulating waveform.

Return Format The query returns an integer.

Example :MOD:AM:INTernal:FREQuency 100

[[:SOURce[<n>]]:MOD:AM:INTernal:FUNCTion [:SOURce[<n>]]:MOD:FM:INTernal:FUNCTion

Syntax [[:SOURce[<n>]]:MOD:AM:INTernal:FUNCTion <wave>
[:SOURce[<n>]]:MOD:AM:INTernal:FUNCTion?
[:SOURce[<n>]]:MOD:FM:INTernal:FUNCTion <wave>
[:SOURce[<n>]]:MOD:FM:INTernal:FUNCTion?

Description Set or query the modulating waveform of AM or FM of the specified source channel.

Parameter

Name	Type	Range	Default
[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
<wave>	Discrete	{SINusoid SQUare RAMP NOISe}	SINusoid

Explanation

- Sine (SINusoid), Square (SQUare), Ramp (RAMP) or Noise (NOISe) can be selected as the modulating waveform. You can set the frequency of the modulating waveform by sending [\[:SOURce\[<n>\]\]:MOD:AM:INTernal:FREQuency](#) or [\[:SOURce\[<n>\]\]:MOD:FM:INTernal:FREQuency](#).
- Sending the [\[:SOURce\[<n>\]\]:MOD:TYPe](#) command can set the modulation type.
- AM: amplitude modulation. The amplitude of the carrier waveform varies with the amplitude of the modulating waveform.
FM: frequency modulation. The frequency of the carrier waveform varies with the amplitude of the modulating waveform.

Return Format The query returns SIN, SQU, RAMP or NOIS.

Example :MOD:AM:INTernal:FUNCTion SQUare

[[:SOURce[<n>]]:MOD:FM[:DEVIation]

Syntax [[:SOURce[<n>]]:MOD:FM[:DEVIation] <dev>
[:SOURce[<n>]]:MOD:FM[:DEVIation]?

Description Set or query the FM frequency deviation of the specified source channel. The default unit is Hz.

Parameter

Name	Type	Range	Default
[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
<dev>	Real	0Hz to the carrier waveform frequency	10kHz

Explanation

- Sending [\[:SOURce\[<n>\]\]:FREQuency\[:FIXed\]](#) can set the carrier waveform frequency and sending [\[:SOURce\[<n>\]\]:MOD:FM:INTernal:FREQuency](#) can set the FM modulation frequency.
- When the modulating waveform amplitude reaches its maximum, the carrier waveform frequency increases by the "frequency deviation". When the modulating waveform amplitude reaches its minimum, the carrier waveform frequency decreases by the "frequency deviation".

Return Format The query returns an integer.

Example :MOD:FM 100

[:SOURce[<n>]]:APPLY?**Syntax** [:SOURce[<n>]]:APPLY?**Description** Query the output configurations of the specified source channel.**Parameter**

Name	Type	Range	Default
[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default

Return Format The query returns the output configurations in "<waveform name>,<frequency>,<amplitude>,<offset>,<start phase>" format. If the corresponding parameter does not exist, it will be replaced by "DEF".**Example** :APPLY? /*The query returns SIN,1000.000000,1.000000,0.000000,0.000000*/**[:SOURce[<n>]]:APPLY:NOISe****[:SOURce[<n>]]:APPLY:PULSe****[:SOURce[<n>]]:APPLY:RAMP****[:SOURce[<n>]]:APPLY:SINusoid****[:SOURce[<n>]]:APPLY:SQUare****[:SOURce[<n>]]:APPLY:USER****Syntax** [:SOURce[<n>]]:APPLY:NOISe [<amp>[,<offset>]]

[:SOURce[<n>]]:APPLY:PULSe [<freq>[,<amp>[,<offset>[,<phase>]]]]

[:SOURce[<n>]]:APPLY:RAMP [<freq>[,<amp>[,<offset>[,<phase>]]]]

[:SOURce[<n>]]:APPLY:SINusoid [<freq>[,<amp>[,<offset>[,<phase>]]]]

[:SOURce[<n>]]:APPLY:SQUare [<freq>[,<amp>[,<offset>[,<phase>]]]]

[:SOURce[<n>]]:APPLY:USER [<freq>[,<amp>[,<offset>[,<phase>]]]]

Description Configure the specified source channel to output the signal with the specified waveform and parameters.**Parameter**

Name	Type	Range	Default
[<n>]	Discrete	{1 2}	If omitted, the operation is performed on source 1 by default
<freq>	Real	Sine: 0.1Hz to 25MHz Square: 0.1Hz to 15MHz Pulse: 0.1Hz to 1MHz Ramp: 0.1Hz to 100kHz Arbitrary waveform: 0.1Hz to 10MHz	100kHz
<amp>	Real	Related to the output impedance: HighZ: 20mVpp to 5Vpp 50Ω: 10mVpp to 2.5Vpp	1Vpp
<offset>	Real	Related to the output impedance and amplitude: HighZ: ±(2.5V-current amplitude/2) 50Ω: ±(1.25V-current amplitude/2)	0mV
<phase>	Real	0° to 360°	0°

- Explanation**
- These commands are used to select the waveform shape.
NOISe: select Noise
PULSe: select Pulse
RAMP: select Ramp
SINusoid: select Sine
SQUare: select Square
USER: select Arbitrary waveform
 - <freq>: set the frequency of the specified waveform (Noise does not have this parameter). The default unit is Hz
<amp>: set the amplitude of the specified waveform. The default unit is Vpp
<offset>: set the DC offset of the specified waveform. The default unit is V
<phase>: set the start phase of the specified waveform (Noise does not have this parameter). The default unit is degree (°).
 - Users can omit one or multiple parameters. When all of the parameters are omitted, the commands only select the waveform shape but do not modify the parameters.
 - The four parameters (<freq>, <amp>, <offset> and <phase>) are in order. You can not omit one parameter and set the one/ones behind it. For example, you can not omit <freq> and set <amp> directly.

:SYSTem Commands

The :SYSTem commands are used to set the system-related parameters, such as the sound and language.

Command List:

- ◆ [:SYSTem:AUToscale](#)
- ◆ [:SYSTem:BEEPer](#)
- ◆ [:SYSTem:ERRor\[:NEXT\]?](#)
- ◆ [:SYSTem:GPIB](#)
- ◆ [:SYSTem:LANGuage](#)
- ◆ [:SYSTem:LOCKed](#)
- ◆ [:SYSTem:PON](#)
- ◆ [:SYSTem:OPTion:INSTall](#)
- ◆ [:SYSTem:OPTion:UNINSTall](#)

:SYSTem:AUToscale

Syntax :SYSTem:AUToscale <bool>

:SYSTem:AUToscale?

Description Enable or disable the **AUTO** key at the front panel, or query the status of the **AUTO** key.

Parameter	Name	Type	Range	Default
<bool>		Bool	{{0 OFF}} {{1 ON}}	1 ON

- Explanation**
- Users can disable the **AUTO** key by sending this command or pressing **Utility** → **Auto Options** → **Lock** and enable it by sending this command.
 - After the **AUTO** key is disabled, the Auto Scale operation is invalid. The [:AUToscale](#) command will be invalid.

Return Format The query returns 0 or 1.

Example :SYSTem:AUToscale ON

:SYSTem:BEEPer

Syntax :SYSTem:BEEPer <bool>

:SYSTem:BEEPer?

Description Enable or disable the beeper, or query the status of the beeper.

Parameter	Name	Type	Range	Default
<bool>		Bool	{{0 OFF}} {{1 ON}}	0 OFF

Return Format The query returns 0 or 1.

Example :SYSTem:BEEPer ON

:SYSTem:ERRor[:NEXT]?

Syntax :SYSTem:ERRor[:NEXT]?

Description Query and delete the last system error message.

Return Format The query returns the error message in "<message number>,<message content>" format. Wherein, <message number> is an integer and <message content> is a ASCII string. For example, -113,"Undefined header; command cannot be found".

:SYSTem:GPIB

Syntax :SYSTem:GPIB <adr>
:SYSTem:GPIB?

Description Set or query the GPIB address.

Parameter	Name	Type	Range	Default
	<adr>	Integer	1 to 30	2

Explanation The GPIB address is not affected by sending [*RST](#).

Return Format The query returns an integer.

Example :SYSTem:GPIB 7

:SYSTem:LANGuage

Syntax :SYSTem:LANGuage <lang>
:SYSTem:LANGuage?

Description Set or query the system language.

Parameter	Name	Type	Range	Default
	<lang>	Discrete	{SCHinese ENGLISH}	SCHinese

Explanation The system language is not affected by sending [*RST](#).

Return Format The query returns SCH or ENGL.

Example :SYSTem:LANGuage SCHinese

:SYSTem:LOCKed

Syntax :SYSTem:LOCKed <bool>
:SYSTem:LOCKed?

Description Enable or disable the keyboard lock function, or query the status of the keyboard lock function.

Parameter	Name	Type	Range	Default
	<bool>	Bool	{{0 OFF}} {{1 ON}}	0 OFF

Return Format The query returns 0 or 1.

:SYSTem:PON

Syntax :SYSTem:PON <pon>

:SYSTem:PON?

Description Set or query the system configuration to be recalled when the oscilloscope is powered on again after power-off.

Parameter	Name	Type	Range	Default
	<pon>	Discrete	{LATest DEFault}	LATest

Return Format The query returns LAT or DEF.

Example :SYSTem:PON LATest

:SYSTem:OPTion:INSTall

Syntax :SYSTem:OPTion:INSTall <license>

Description Install the option.

Parameter	Name	Type	Range	Default
	<license>	ASCII String	Option serial number supplied by RIGOL No spaces between characters.	None

Example :SYSTem:OPTion:INSTall PDUY9N9QTS9PQSWPLAETRD3UJHYA

:SYSTem:OPTion:UNINSTall

Syntax :SYSTem:OPTion:UNINSTall

Description Uninstall the options installed.

:TIMEbase Commands

The :TIMEbase commands are used to set the horizontal parameters, such as enabling the delayed sweep and setting the horizontal timebase mode.

Command List:

- ◆ [:TIMEbase:DElay:ENABle](#)
- ◆ [:TIMEbase:DElay:OFFSet](#)
- ◆ [:TIMEbase:DElay:SCALe](#)
- ◆ [:TIMEbase\[:MAIN\]:OFFSet](#)
- ◆ [:TIMEbase\[:MAIN\]:SCALe](#)
- ◆ [:TIMEbase:MODE](#)

:TIMEbase:DElay:ENABle

Syntax :TIMEbase:DElay:ENABle <bool>

:TIMEbase:DElay:ENABle?

Description Enable or disable the delayed sweep, or query the status of the delayed sweep.

Parameter	Name	Type	Range	Default
	<bool>	Bool	{{0 OFF} {1 ON}}	0 OFF

Explanation Delayed sweep can be used to enlarge a length of waveform horizontally to view the waveform details.

Return Format The query returns 0 or 1.

Example :TIMEbase:DElay:ENABle ON

:TIMEbase:DElay:OFFSet

Syntax :TIMEbase:DElay:OFFSet <offset>

:TIMEbase:DElay:OFFSet?

Description Set or query the delayed timebase offset. The default unit is s.

Parameter	Name	Type	Range	Default
	<offset>	Real	-(LeftTime - DelayRange/2) to (RightTime - DelayRange/2)	0

Explanation LeftTime = $6 \times \text{MainScale} - \text{MainOffset}$
 RightTime = $6 \times \text{MainScale} + \text{MainOffset}$
 DelayRange = $12 \times \text{DelayScale}$

Return Format The query returns the offset in scientific notation.

Example :TIMEbase:DElay:OFFSet 0.000002 /*Set the delayed timebase offset to $2\mu\text{s}$ */
 :TIMEbase:DElay:OFFSet? /*The query returns $2.0000000\text{e-}06$ */

Related Commands [:TIMEbase:DElay:SCALe](#)
[:TIMEbase\[:MAIN\]:OFFSet](#)
[:TIMEbase\[:MAIN\]:SCALe](#)

:TIMebase:DELay:SCALE

Syntax :TIMebase:DELay:SCALE <scale_value>

:TIMebase:DELay:SCALE?

Description Set or query the delayed timebase scale. The default unit is s/div.

Parameter	Name	Type	Range	Default
	<scale_value>	Real	See Explanation	500ns

- Explanation**
- The maximum of <scale_value> is the main timebase scale currently set and the minimum is related to the current sample rate (satisfying the inequality: current sample rate * delayed timebase scale * 7 ≥ 14) and is no less than 1ns.
 - The delayed timebase scale can only be the maximum or the value decreased from the maximum in 1-2-5 step. If the minimum calculated by the inequality mentioned above is not a settable value, the larger settable value that is nearest to the calculated value will be used as the minimum. If the value set is between two settable values, the smaller one will be used as the delayed timebase scale automatically.
 - For example, when the main timebase scale currently set is 10.00μs and the sample rate is 500MSa/s, the maximum of <scale_value> is 10.00μs. The minimum is calculated using the inequality: $\geq 14 / (500\text{MSa/s} * 7) = 40\text{ns}$. Taking the larger settable value that is nearest to the calculated value, the minimum is 50ns. At this point, if 0.000009 (namely 9μs) is set, the instrument sets the delayed timebase scale to 5μs.

Return Format The query returns the scale in scientific notation.

Example :TIMebase:DELay:SCALE 0.00000005 /*Set the delayed timebase scale to 50ns*/
:TIMebase:DELay:SCALE? /*The query returns 5.0000000e-08*/

Related Command [:TIMebase\[:MAIN\]:SCALE](#)

:TIMebase[:MAIN]:OFFSet

Syntax :TIMebase[:MAIN]:OFFSet <offset>

:TIMebase[:MAIN]:OFFSet?

Description Set or query the main timebase offset. The default unit is s.

Parameter	Name	Type	Range	Default
	<offset>	Real	-Screen/2 to 1s or -Screen/2 to 5000s	0

- Explanation**
- This command is invalid when the horizontal timebase mode is Roll and the oscilloscope is in the Run state.
 - When the horizontal timebase mode is YT and the horizontal timebase is 20ms or larger (namely the "Slow Sweep" mode), this command is invalid when the oscilloscope is in the Stopping state.

Return Format The query returns the offset in scientific notation.

Example :TIMebase:MAIN:OFFSet 0.0002 /*Set the main timebase offset to 20ms*/

Related Commands [:TIMebase:MODE](#)
[:TIMebase\[:MAIN\]:SCALE](#)

[:RUN](#)

[:STOP](#)

:TIMebase[:MAIN]:SCALE

Syntax :TIMebase[:MAIN]:SCALE <scale_value>

:TIMebase[:MAIN]:SCALE?

Description Set or query the main timebase scale. The default unit is s/div.

Parameter	Name	Type	Range	Default
	<scale_value>	Real	Normal: 5ns to 50s in 1-2-5 step Roll: 200ms to 50s in 1-2-5 step	1 μ s

Explanation When the horizontal timebase mode is YT and the horizontal timebase is 20ms or larger (namely the "Slow Sweep" mode), this command is invalid when the oscilloscope in the Stopping state.

Return Format The query returns the main timebase scale in scientific notation.

Example :TIMebase:MAIN:SCALE 0.0002 /*Set the main timebase scale to 200 μ s*/
:TIMebase:MAIN:SCALE? /*The query returns 2.0000000e-04*/

Related Command [:TIMebase:MODE](#)

:TIMebase:MODE

Syntax :TIMebase:MODE <mode>

:TIMebase:MODE?

Description Set or query the mode of the horizontal timebase.

Parameter	Name	Type	Range	Default
	<mode>	Discrete	{MAIN XY ROLL}	MAIN

Return Format The query returns MAIN, XY or ROLL.

Example :TIMebase:MODE XY

Related Commands [:TIMebase:DElay:OFFSet](#)
[:TIMebase:DElay:SCALE](#)
[:TIMebase\[:MAIN\]:OFFSet](#)
[:TIMebase\[:MAIN\]:SCALE](#)

:TRIGger Commands

The :TRIGger commands are used to set the trigger system of the oscilloscope.

Command List:

- ◆ [:TRIGger:MODE](#)
- ◆ [:TRIGger:COUPling](#)
- ◆ [:TRIGger:STATus?](#)
- ◆ [:TRIGger:SWEep](#)
- ◆ [:TRIGger:HOLDoff](#)
- ◆ [:TRIGger:NREJect](#)
- ◆ [:TRIGger:EDGE](#)
- ◆ [:TRIGger:PULSe](#)
- ◆ [:TRIGger:SLOPe](#)
- ◆ [:TRIGger:VIDeo](#)
- ◆ [:TRIGger:PATTErn](#)
- ◆ [:TRIGger:DURATIon](#)
- ◆ [:TRIGger:TIMEout \(Option\)](#)
- ◆ [:TRIGger:RUNT \(Option\)](#)
- ◆ [:TRIGger:WINDows \(Option\)](#)
- ◆ [:TRIGger:DELAy \(Option\)](#)
- ◆ [:TRIGger:SHOLd](#)
- ◆ [:TRIGger:NEDGE \(Option\)](#)
- ◆ [:TRIGger:RS232 \(Option\)](#)
- ◆ [:TRIGger:IIC \(Option\)](#)
- ◆ [:TRIGger:SPI \(Option\)](#)

:TRIGger:MODE

Syntax :TRIGger:MODE <mode>

:TRIGger:MODE?

Description Select or query the trigger type.

Parameter	Name	Type	Range	Default
	<mode>	Discrete	{EDGE PULSe RUNT WIND NEDG SLOPe VIDeo PATTErn DELAy TIMEout DURation SHOLd RS232 IIC SPI}	EDGE

Return Format The query returns EDGE, PULS, RUNT, WIND, NEDG, SLOP, VID, PATT, DEL, TIM, DUR, SHOL, RS232, IIC or SPI.

Example :TRIGger:MODE SLOPe

:TRIGger:COUPLing

Syntax :TRIGger:COUPLing <couple>

:TRIGger:COUPLing?

Description Select or query the trigger coupling type.

Parameter	Name	Type	Range	Default
	<couple>	Discrete	{AC DC LFReject HFReject}	DC

- Explanation**
- This command is only applicable to the edge trigger.
 - AC: block all the DC components and attenuate signals lower than 75 kHz.
 - DC: allow DC and AC components into the trigger path.
 - LFReject: block the DC components and reject the low frequency components (lower than 75 kHz).
 - HFReject: reject the high frequency components (higher than 75 kHz).

Return Format The query returns AC, DC, LFR or HFR.

Example :TRIGger:COUPLing LFReject

:TRIGger:STATus?

Syntax :TRIGger:STATus?

Description Query the current trigger status.

Return Format The query returns TD, WAIT, RUN, AUTO or STOP.

:TRIGger:SWEep

Syntax :TRIGger:SWEep <sweep>

:TRIGger:SWEep?

Description Set or query the trigger mode.

Parameter	Name	Type	Range	Default
	<sweep>	Discrete	{AUTO NORMal SINGle}	AUTO

- Explanation**
- AUTO: auto trigger. No matter whether the trigger condition is met, there is always waveform display.
 - NORMal: normal trigger. Display waveform when the trigger condition is met; otherwise, the oscilloscope holds the original waveform and waits for the next trigger.
 - SINGle: single trigger. The oscilloscope waits for a trigger and displays the waveform when the trigger condition is met and then stops.

Return Format The query returns AUTO, NORM or SING.

Example :TRIGger:SWEep SINGle

:TRIGger:HOLDoff

Syntax :TRIGger:HOLDoff <value>

:TRIGger:HOLDoff?

Description Set or query the trigger holdoff time. The default unit is s.

Parameter	Name	Type	Range	Default
	<value>	Real	16ns to 10s	16ns

- Explanation**
- Trigger holdoff can be used to stably trigger the complex waveforms (such as pulse series). Holdoff time is the time that the oscilloscope waits before re-arming the trigger circuitry. The oscilloscope will not trigger until the holdoff time expires.
 - When the trigger type is video, RS232, I2C or SPI, this setting will be not available.

Return Format The query returns the trigger holdoff time in scientific notation.

Example :TRIGger:HOLDoff 0.0000002 /*Set the trigger holdoff time to 200ns*/
:TRIGger:HOLDoff? /*The query returns 2.000000e-07*/

:TRIGger:NREject

Syntax :TRIGger:NREject <bool>

:TRIGger:NREject?

Description Enable or disable the noise rejection, or query the status of the noise rejection.

Parameter	Name	Type	Range	Default
	<bool>	Bool	{{0 OFF}} {1 ON}}	0 OFF

Explanation Noise rejection reduces the possibility of noise trigger.

Return Format The query returns 0 or 1.

Example :TRIGger:NREject ON

:TRIGger:EDGE

Command List:

- ◆ [:TRIGger:EDGE:SOURce](#)
- ◆ [:TRIGger:EDGE:SLOPe](#)
- ◆ [:TRIGger:EDGE:LEVel](#)

:TRIGger:EDGE:SOURce

Syntax :TRIGger:EDGE:SOURce <source>

:TRIGger:EDGE:SOURce?

Description Set or query the trigger source in edge trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4 AC}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3, CHAN4 or AC.

Example :TRIGger:EDGE:SOURce CHANnel1

:TRIGger:EDGE:SLOPe

Syntax :TRIGger:EDGE:SLOPe <slope>

:TRIGger:EDGE:SLOPe?

Description Set or query the edge type in edge trigger.

Parameter	Name	Type	Range	Default
	<slope>	Discrete	{POSitive NEGative RFALI}	POSitive

Return Format The query returns POS, NEG or RFAL.

Example :TRIGger:EDGE:SLOPe NEGative

Related Command [:TRIGger:MODE](#)

:TRIGger:EDGE:LEVEL

Syntax :TRIGger:EDGE:LEVEL <level>

:TRIGger:EDGE:LEVEL?

Description Set or query the trigger level in edge trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<level>	Real	($\pm 5 \times \text{VerticalScale}$ from the screen center) - OFFSET	0

Explanation For VerticalScale, refer to the [:CHANnel<n>:SCALE](#) command. For OFFSET, refer to the [:CHANnel<n>:OFFSET](#) command.

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

Example :TRIGger:EDGE:LEVEL 0.16

Related Commands [:CHANnel<n>:SCALE](#)
[:CHANnel<n>:OFFSET](#)

:TRIGger:PULSe

Command List:

- ◆ [:TRIGger:PULSe:SOURce](#)
- ◆ [:TRIGger:PULSe:WHEN](#)
- ◆ [:TRIGger:PULSe:WIDTh](#)
- ◆ [:TRIGger:PULSe:UWIDth](#)
- ◆ [:TRIGger:PULSe:LWIDth](#)
- ◆ [:TRIGger:PULSe:LEVel](#)

:TRIGger:PULSe:SOURce

Syntax :TRIGger:PULSe:SOURce <source>
:TRIGger:PULSe:SOURce?

Description Set or query the trigger source in pulse width trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:PULSe:SOURce CHANnel1 /*Set the trigger source to CH1*/

:TRIGger:PULSe:WHEN

Syntax :TRIGger:PULSe:WHEN <when>
:TRIGger:PULSe:WHEN?

Description Set or query the trigger condition in pulse width trigger.

Parameter	Name	Type	Range	Default
	<when>	Discrete	{PGReater PLESs NGReater NLESs PGLess NGLess}	PGReater

- Explanation**
- PGReater/PLESs: you need to specify a pulse width (refer to [:TRIGger:PULSe:WIDTh](#)). The oscilloscope triggers when the positive pulse width of the input signal is greater/lower than the specified Pulse Width.
 - NGReater/NLESs: you need to specify a pulse width (refer to [:TRIGger:PULSe:WIDTh](#)). The oscilloscope triggers when the negative pulse width of the input signal is greater/lower than the specified Pulse Width.
 - PGLess/NGLess: you need to specify an upper (refer to [:TRIGger:PULSe:UWIDth](#)) and a lower (refer to [:TRIGger:PULSe:LWIDth](#)) pulse width. The oscilloscope triggers when the positive/negative pulse width of the input signal is greater than the specified lower pulse width and lower than the upper pulse width.

Return Format The query returns PGR, PLES, NGR, NLES, PGL or NGL.

Example :TRIGger:PULSe:WHEN NLESs /*Set the trigger condition to NLESs*/

Related Commands [:TRIGger:PULSe:WIDTh](#)
[:TRIGger:PULSe:UWIDth](#)
[:TRIGger:PULSe:LWIDth](#)

:TRIGger:PULSe:WIDTh

Syntax :TRIGger:PULSe:WIDTh <width>

:TRIGger:PULSe:WIDTh?

Description Set or query the pulse width in pulse width trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<width>	Real	8ns to 10s	PGReater, NGReater: 1μs PLESSs, NLESSs: 2μs

Explanation This command is available when the trigger condition (refer to [:TRIGger:PULSe:WHEN](#)) is PGReater, PLESSs, NGReater and NLESSs.

Return Format The query returns the pulse width in scientific notation.

Example :TRIGger:PULSe:WIDTh 0.000003 /*Set the pulse width to 3μs*/
:TRIGger:PULSe:WIDTh? /*The query returns3.000000e-06*/

Related Command [:TRIGger:PULSe:WHEN](#)

:TRIGger:PULSe:UWIDTh

Syntax :TRIGger:PULSe:UWIDTh <width>

:TRIGger:PULSe:UWIDTh?

Description Set or query the upper pulse width in pulse width trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<width>	Real	16ns to 10s	1μs

Explanation This command is available when the trigger condition (refer to [:TRIGger:PULSe:WHEN](#)) is PGLess and NGLess.

Return Format The query returns the upper pulse width in scientific notation.

Example :TRIGger:PULSe:UWIDTh 0.000003 /*Set the upper pulse width to 3μs*/
:TRIGger:PULSe:UWIDTh? /*The query returns3.000000e-06*/

Related Commands [:TRIGger:PULSe:WHEN](#)
[:TRIGger:PULSe:LWIDTh](#)

:TRIGger:PULSe:LWIDth

Syntax :TRIGger:PULSe:LWIDth <width>

:TRIGger:PULSe:LWIDth?

Description Set or query the lower pulse width in pulse width trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<width>	Real	8ns to 9.99s	992ns

Explanation This command is available when the trigger condition (refer to [:TRIGger:PULSe:WHEN](#)) is PGLess and NGLess.

Return Format The query returns the lower pulse width in scientific notation.

Example :TRIGger:PULSe:LWIDth 0.000003 /*Set the lower pulse width to 3μs*/
:TRIGger:PULSe:LWIDth? /*The query returns 3.000000e-06*/

Related Commands [:TRIGger:PULSe:WHEN](#)
[:TRIGger:PULSe:UWIDth](#)

:TRIGger:PULSe:LEVel

Syntax :TRIGger:PULSe:LEVel <level>

:TRIGger:PULSe:LEVel?

Description Set or query the trigger level in pulse width trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<level>	Real	(± 5 × VerticalScale from the screen center) - OFFSet	0

Explanation For VerticalScale, refer to the [:CHANnel<n>:SCALE](#) command. For OFFSet, refer to the [:CHANnel<n>:OFFSet](#) command.

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

Example :TRIGger:PULSe:LEVel 0.16 /*Set the trigger level to 160mV*/
:TRIGger:PULSe:LEVel? /*The query returns 1.600000e-01*/

Related Commands [:CHANnel<n>:SCALE](#)
[:CHANnel<n>:OFFSet](#)

:TRIGger:SLOPe

Command List:

- ◆ [:TRIGger:SLOPe:SOURce](#)
- ◆ [:TRIGger:SLOPe:WHEN](#)
- ◆ [:TRIGger:SLOPe:TIME](#)
- ◆ [:TRIGger:SLOPe:TUPPer](#)
- ◆ [:TRIGger:SLOPe:TLOWer](#)
- ◆ [:TRIGger:SLOPe:WINDow](#)
- ◆ [:TRIGger:SLOPe:ALEVel](#)
- ◆ [:TRIGger:SLOPe:BLEVel](#)

:TRIGger:SLOPe:SOURce

Syntax :TRIGger:SLOPe:SOURce <source>
:TRIGger:SLOPe:SOURce?

Description Set or query the trigger source in slope trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:SLOPe:SOURce CHANnel2

:TRIGger:SLOPe:WHEN

Syntax :TRIGger:SLOPe:WHEN <when>

:TRIGger:SLOPe:WHEN?

Description Set or query the trigger condition in slope trigger.

Parameter	Name	Type	Range	Default
	<when>	Discrete	{PGReater PLESs NGReater NLESs PGLess NGLess}	PGReater

- Explanation**
- PGReater/PLESs: you need to specify a time value (refer to [:TRIGger:SLOPe:TIME](#)). The oscilloscope triggers when the positive slope time of the input signal is greater/lower than the specified time.
 - NGReater/NLESs: you need to specify a time value (refer to [:TRIGger:SLOPe:TIME](#)). The oscilloscope triggers when the negative slope time of the input signal is greater/lower than the specified time.
 - PGLess/NGLess: you need to specify an upper limit (refer to [:TRIGger:SLOPe:TUPPer](#)) and a lower limit (refer to [:TRIGger:SLOPe:TLOWer](#)) of the time. The oscilloscope triggers when the positive/negative slope time of the input signal is greater than the specified lower limit and lower than the specified upper limit.

Return Format The query returns PGR, PLES, NGR, NLES, PGL or NGL.

Example :TRIGger:SLOPe:WHEN NLESs /*Set the trigger condition to NLESs*/

Related Commands [:TRIGger:SLOPe:TIME](#)
[:TRIGger:SLOPe:TUPPer](#)
[:TRIGger:SLOPe:TLOWer](#)

:TRIGger:SLOPe:TIME

Syntax :TRIGger:SLOPe:TIME <time>

:TRIGger:SLOPe:TIME?

Description Set or query the time value in slope trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<time>	Real	8ns to 10s	PGReater, NGReater: 1μs PLESs, NLESs: 2μs

Explanation This command is available when the trigger condition (refer to [:TRIGger:SLOPe:WHEN](#)) is PGReater, PLESs, NGReater and NLESs.

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

Example :TRIGger:SLOPe:TIME 0.000003 /*Set the time value to 3μs*/
:TRIGger:SLOPe:TIME? /*The query returns 3.000000e-06*/

Related Command [:TRIGger:SLOPe:WHEN](#)

:TRIGger:SLOPe:TUPPer

Syntax :TRIGger:SLOPe:TUPPer <time>

:TRIGger:SLOPe:TUPPer?

Description Set or query the upper limit of the time in slope trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<time>	Real	16ns to 10s	1 μ s

Explanation This command is available when the trigger condition (refer to [:TRIGger:SLOPe:WHEN](#)) is PGLess and NGLess.

Return Format The query returns the upper limit of the time in scientific notation.

Example :TRIGger:SLOPe:TUPPer 0.000003 /*Set the upper limit of the time to 3 μ s*/
:TRIGger:SLOPe:TUPPer? /*The query returns 3.000000e-06*/

Related Commands [:TRIGger:SLOPe:WHEN](#)
[:TRIGger:SLOPe:TLOWer](#)

:TRIGger:SLOPe:TLOWer

Syntax :TRIGger:SLOPe:TLOWer <time>

:TRIGger:SLOPe:TLOWer?

Description Set or query the lower limit of the time in slope trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<time>	Real	8ns to 9.99s	992ns

Explanation This command is available when the trigger condition (refer to [:TRIGger:SLOPe:WHEN](#)) is PGLess and NGLess.

Return Format The query returns the lower limit of the time in scientific notation.

Example :TRIGger:SLOPe:TLOWer 0.000000020 /*Set the lower limit of the time to 20ns*/
:TRIGger:SLOPe:TUPPer? /*The query returns 2.000000e-08*/

Related Commands [:TRIGger:SLOPe:WHEN](#)
[:TRIGger:SLOPe:TUPPer](#)

:TRIGger:SLOPe:WINDow

Syntax :TRIGger:SLOPe:WINDow <window>

:TRIGger:SLOPe:WINDow?

Description Set or query the vertical window type in slope trigger.

Parameter	Name	Type	Range	Default
	<window>	Discrete	{TA TB TAB}	TA

- Explanation**
- TA: only adjust the upper limit of the trigger level, refer to [:TRIGger:SLOPe:ALEVel](#).
 - TB: only adjust the lower limit of the trigger level, refer to [:TRIGger:SLOPe:BLEVel](#).
 - TAB: adjust the upper and lower limits of the trigger level at the same time, refer to [:TRIGger:SLOPe:ALEVel](#) and [:TRIGger:SLOPe:BLEVel](#).

Return Format The query returns TA, TB or TAB.

Example :TRIGger:SLOPe:WINDow TB

Related Commands [:TRIGger:SLOPe:ALEVel](#)
[:TRIGger:SLOPe:BLEVel](#)

:TRIGger:SLOPe:ALEVel

Syntax :TRIGger:SLOPe:ALEVel <level>

:TRIGger:SLOPe:ALEVel?

Description Set or query the upper limit of the trigger level in slope trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<level>	Real	($\pm 5 \times \text{VerticalScale}$ from the screen center) - OFFSet	2V

Explanation For VerticalScale, refer to [:CHANnel<n>:SCALE](#). For OFFSet, refer to [:CHANnel<n>:OFFSet](#).

Return Format The query returns the upper limit of the trigger level in scientific notation.

Example :TRIGger:SLOPe:ALEVel 0.16 /*Set the upper limit of the trigger level to 160mV*/
:TRIGger:SLOPe:ALEVel /*The query returns 1.600000e-01*/

Related Commands [:CHANnel<n>:SCALE](#)
[:CHANnel<n>:OFFSet](#)
[:TRIGger:SLOPe:BLEVel](#)

:TRIGger:SLOPe:BLEVel

Syntax :TRIGger:SLOPe:BLEVel <level>

:TRIGger:SLOPe:BLEVel?

Description Set or query the lower limit of the trigger level in slope trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<level>	Real	($\pm 5 \times \text{VerticalScale}$ from the screen center) - OFFSet	0

Explanation For VerticalScale, refer to [:CHANnel<n>:SCALE](#). For OFFSet, refer to [:CHANnel<n>:OFFSet](#).

Return The query returns the lower limit of the trigger level in scientific notation.

Format

Example :TRIGger:SLOPe:BLEVel 0.16 /*Set the lower limit of the trigger level to 160mV*/
:TRIGger:SLOPe:BLEVel /*The query returns 1.600000e-01*/

Related [:CHANnel<n>:SCALE](#)

Commands [:CHANnel<n>:OFFSet](#)

[:TRIGger:SLOPe:ALEVel](#)

:TRIGger:VIDeo

Command List:

- ◆ [:TRIGger:VIDeo:SOURce](#)
- ◆ [:TRIGger:VIDeo:POLarity](#)
- ◆ [:TRIGger:VIDeo:MODE](#)
- ◆ [:TRIGger:VIDeo:LINE](#)
- ◆ [:TRIGger:VIDeo:STANdard](#)
- ◆ [:TRIGger:VIDeo:LEVel](#)

:TRIGger:VIDeo:SOURce

Syntax :TRIGger:VIDeo:SOURce <source>

:TRIGger:VIDeo:SOURce?

Description Select or query the trigger source in video trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:VIDeo:SOURce CHANnel2

:TRIGger:VIDeo:POLarity

Syntax :TRIGger:VIDeo:POLarity <polarity>

:TRIGger:VIDeo:POLarity?

Description Select or query the video polarity in video trigger.

Parameter	Name	Type	Range	Default
	<polarity>	Discrete	{POSitive NEGative}	POSitive

Return Format The query returns POS or NEG.

Example :TRIGger:VIDeo:POLarity POSitive

:TRIGger:VIDeo:MODE

Syntax :TRIGger:VIDeo:MODE <mode>
:TRIGger:VIDeo:MODE?

Description Set or query the sync type in video trigger.

Parameter	Name	Type	Range	Default
	<mode>	Discrete	{ODDField EVENfield LINE ALINes}	ALINes

- Explanation**
- ODDField: trigger on the rising edge of the first ramp waveform pulse in the odd field. This type is available when the video standard is NTSC or PAL/SECAM.
 - EVENfield: trigger on the rising edge of the first ramp waveform pulse in the even field. This type is available when the video standard is NTSC or PAL/SECAM.
 - LINE: for the NTSC and PAL/SECAM video standards, trigger on the specified line in the odd or even field; for the 480P and 576P video standards, trigger on the specified line.
 - ALINes: trigger on all the horizontal sync pulses.

Return Format The query returns ODDF, EVEN, LINE or ALIN.

Example :TRIGger:VIDeo:MODE ODDField

Related Commands [:TRIGger:VIDeo:LINE](#)
[:TRIGger:VIDeo:STANdard](#)

:TRIGger:VIDeo:LINE

Syntax :TRIGger:VIDeo:LINE <line>
:TRIGger:VIDeo:LINE?

Description Set or query the line number when the sync type in video trigger is LINE.

Parameter	Name	Type	Range	Default
	<line>	Integer	Refer to Explanation	1

- Explanation**
- NTSC: 1 to 525
 - PAL/SECAM: 1 to 625
 - 480P: 1 to 525
 - 576P: 1 to 625

Return Format The query returns an integer.

Example :TRIGger:VIDeo:LINE 100

Related Command [:TRIGger:VIDeo:MODE](#)

:TRIGger:VIDeo:STANdard

Syntax :TRIGger:VIDeo:STANdard <standard>

:TRIGger:VIDeo:STANdard?

Description Set or query the video standard in video trigger.

Parameter	Name	Type	Range	Default
	<standard>	Discrete	{PALSecam NTSC 480P 576P}	NTSC

- Explanation**
- PALSecam:
 - PAL: the frame frequency is 25 frames per second. The TV sweep line is 625 with the odd field goes first and the even field follows behind.
 - Secam: the frame frequency is 25 frames per second. The sweep line is 625 with interlacing sweep.
 - NTSC: the field frequency is 60 fields per second and the frame frequency is 30 frames per second. The TV sweep line is 525 with the even field goes first and the odd field follows behind.
 - 480P: the frame frequency is 60 frames per second; the TV sweep line is 525; line-by-line sweep; the line frequency is 31.5 kHz.
 - 576P: the frame frequency is 60 frames per second; the TV sweep line is 625; line-by-line sweep.

Return Format The query returns PALS, NTSC, 480P or 576P.

Example :TRIGger:VIDeo:STANdard NTSC

Related Commands [:TRIGger:VIDeo:LINE](#)
[:TRIGger:VIDeo:MODE](#)

:TRIGger:VIDeo:LEVel

Syntax :TRIGger:VIDeo:LEVel <level>

:TRIGger:VIDeo:LEVel?

Description Set or query the trigger level in video trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<level>	Real	($\pm 5 \times \text{VerticalScale}$ from the screen center) - OFFSet	0

- Explanation**
- For VerticalScale, refer to [:CHANnel<n>:SCALE](#).
 - For OFFSet, refer to [:CHANnel<n>:OFFSet](#).

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

Example :TRIGger:VIDeo:LEVel 0.16 /*Set the trigger level to 160mV*/
 :TRIGger:VIDeo:LEVel /*The query returns 1.600000e-01*/

Related Commands [:CHANnel<n>:SCALE](#)
[:CHANnel<n>:OFFSet](#)

:TRIGger:PATtern

Command List:

- ◆ [:TRIGger:PATtern:PATtern](#)
- ◆ [:TRIGger:PATtern:LEVel](#)

:TRIGger:PATtern:PATtern

Syntax :TRIGger:PATtern:PATtern <pattern1>,<pattern2>,<pattern3>,<pattern4>
:TRIGger:PATtern:PATtern?

Description Set or query the pattern of each channel in pattern trigger.

Parameter	Name	Type	Range	Default
	<pattern1>	Discrete	{H L X R F}	X
	<pattern2>	Discrete	{H L X R F}	X
	<pattern3>	Discrete	{H L X R F}	X
	<pattern4>	Discrete	{H L X R F}	X

- Explanation**
- H: set the pattern of the channel selected to "H", namely the voltage level is higher than the threshold level of the channel.
 - L: set the pattern of the channel selected to "L", namely the voltage level is lower than the threshold level of the channel.
 - X: set the pattern of the channel selected to "Don't Care", namely this channel is not used as a part of the pattern. When all the four channels in the pattern are set to "Don't Care", the oscilloscope will not trigger.
 - R/F: set the pattern to the rising or falling edge of the channel selected.
 - You can not to specify the patterns of two channels to the rising edge or falling edge at the same time. If one edge item is currently defined and then another edge item is defined in another channel in the pattern, the former edge item defined will be replaced by X.
 - The four parameters from left to right in the command set the patterns of CH1, CH2, CH3 and CH4 respectively.

Return Format The query returns the patterns of all the channels.

Example :TRIGger:PATtern:PATtern H,R,L,X

:TRIGger:PATtern:LEVel

Syntax :TRIGger:PATtern:LEVel <chan>,<level>

:TRIGger:PATtern:LEVel? <chan>

Description Set or query the trigger level of the specified channel in pattern trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<chan>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1
	<level>	Real	($\pm 5 \times \text{VerticalScale}$ from the screen center) - OFFSet	0

Explanation For VerticalScale, refer to [:CHANnel<n>:SCALE](#). For OFFSet, refer to [:CHANnel<n>:OFFSet](#).

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

Example :TRIGger:PATtern:LEVel CHANnel2,0.16 /*Set the trigger level of CH2 to 160mV*/

Related Commands [:CHANnel<n>:SCALE](#)
[:CHANnel<n>:OFFSet](#)

:TRIGger:DURATION

Command List:

- ◆ [:TRIGger:DURATION:SOURce](#)
- ◆ [:TRIGger:DURATION:TYPE](#)
- ◆ [:TRIGger:DURATION:WHEN](#)
- ◆ [:TRIGger:DURATION:TUPPer](#)
- ◆ [:TRIGger:DURATION:TLOWer](#)

:TRIGger:DURATION:SOURce

Syntax :TRIGger:DURATION:SOURce <source>
:TRIGger:DURATION:SOURce?

Description Set or query the trigger source in duration trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:DURATION:SOURce CHANnel2

:TRIGger:DURATION:TYPE

Syntax :TRIGger:DURATION:TYPE <type1>,<type2>,<type3>,<type4>
:TRIGger:DURATION:TYPE?

Description Set or query the pattern of each channel in duration trigger.

Parameter	Name	Type	Range	Default
	<type1>	Discrete	{H L X}	X
	<type2>	Discrete	{H L X}	X
	<type3>	Discrete	{H L X}	X
	<type4>	Discrete	{H L X}	X

- Explanation**
- H: set the pattern of the channel selected to "H", namely the voltage level is higher than the threshold level of the channel.
 - L: set the pattern of the channel selected to "L", namely the voltage level is lower than the threshold level of the channel.
 - X: set the pattern of the channel selected to "Don't Care", namely this channel is not used as a part of the pattern. When all the four channels in the pattern are set to "Don't Care", the oscilloscope will not trigger.
 - The four parameters from left to right in the command set the patterns of CH1, CH2, CH3 and CH4 respectively.

Return Format The query returns the patterns of the four channels.

Example :TRIGger:DURATION:TYPE L,X,H,L

:TRIGger:DURATion:WHEN

Syntax :TRIGger:DURATion:WHEN <when>

:TRIGger:DURATion:WHEN?

Description Set or query the trigger condition in duration trigger.

Parameter	Name	Type	Range	Default
	<when>	Discrete	{GREater LESS GLESS}	GREater

- Explanation**
- GREater: you need to specify a time (refer to [:TRIGger:DURATion:TLOWer](#)). The oscilloscope triggers when the duration of the pattern is greater than the preset time.
 - LESS: you need to specify a time (refer to [:TRIGger:DURATion:TUPPer](#)). The oscilloscope triggers when the duration of the pattern is lower than the preset time.
 - GLESS: you need to specify an upper limit of the time (refer to [:TRIGger:DURATion:TUPPer](#)) and lower limit of the time (refer to [:TRIGger:DURATion:TLOWer](#)). The oscilloscope triggers when the duration of the pattern is lower than the preset upper limit of the time and greater than the preset lower limit of the time.

Return Format The query returns GRE, LESS or GLEs.

Example :TRIGger:DURATion:WHEN LESS

Related Commands [:TRIGger:DURATion:TLOWer](#)
[:TRIGger:DURATion:TUPPer](#)

:TRIGger:DURATion:TUPPer

Syntax :TRIGger:DURATion:TUPPer <NR3>

:TRIGger:DURATion:TUPPer?

Description Set or query the duration time upper limit in duration trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<NR3>	Real	Related to the trigger condition LESS: 8ns to 10s GLESS: 16ns to 10s	2μs

Explanation This command is available when the trigger condition ([:TRIGger:DURATion:WHEN](#)) is LESS or GLEs.

Return Format The query returns the duration time upper limit in scientific notation.

Example :TRIGger:DURATion:TUPPer 0.000003 /*Set the duration time upper limit to 3μs*/
:TRIGger:DURATion:TUPPer? /*The query returns 3.000000e-06*/

Related Commands [:TRIGger:DURATion:WHEN](#)
[:TRIGger:DURATion:TLOWer](#)

:TRIGger:DURATion:TLOWer

Syntax :TRIGger:DURATion:TLOWer <NR3>

:TRIGger:DURATion:TLOWer?

Description Set or query the duration time lower limit in duration trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<NR3>	Real	8ns to 9.99s	1 μ s

Explanation This command is available when the trigger condition ([:TRIGger:DURATion:WHEN](#)) is GREater or GLESS.

Return Format The query returns the duration time lower limit in scientific notation.

Example :TRIGger:DURATion:TLOWer 0.000003 /*Set the duration time lower limit to 3 μ s*/
:TRIGger:DURATion:TLOWer? /*The query returns 3.000000e-06*/

Related Commands [:TRIGger:DURATion:WHEN](#)
[:TRIGger:DURATion:TUPPer](#)

:TRIGger:TIMEout (Option)

Command List:

- ◆ [:TRIGger:TIMEout:SOURce](#)
- ◆ [:TRIGger:TIMEout:SLOPe](#)
- ◆ [:TRIGger:TIMEout:TIME](#)

:TRIGger:TIMEout:SOURce

Syntax :TRIGger:TIMEout:SOURce <source>
:TRIGger:TIMEout:SOURce?

Description Set or query the trigger source in timeout trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:TIMEout:SOURce CHANnel2

:TRIGger:TIMEout:SLOPe

Syntax :TRIGger:TIMEout:SLOPe <slope>
:TRIGger:TIMEout:SLOPe?

Description Set or query the edge type in timeout trigger.

Parameter	Name	Type	Range	Default
	<slope>	Discrete	{POSitive NEGative RFALI}	POSitive

- Explanation**
- POSitive: start timing when the rising edge of the input signal passes through the trigger level.
 - NEGative: start timing when the falling edge of the input signal passes through the trigger level.
 - RFALI: start timing when any edge of the input signal passes through the trigger level.

Return Format The query returns POS, NEG or RFAL.

Example :TRIGger:TIMEout:SLOPe NEGative

:TRIGger:TIMEout:TIME

Syntax :TRIGger:TIMEout:TIME <NR3>

:TRIGger:TIMEout:TIME?

Description Set or query the timeout time in timeout trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<NR3>	Real	16ns to 10s	16ns

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

Example :TRIGger:TIMEout:TIME 0.002 /*Set the timeout time to 2ms*/
:TRIGger:TIMEout:TIME? /*The query returns 2.000000e-03*/

Related Command [:TRIGger:TIMEout:TIME](#)

:TRIGger:RUNT (Option)

Command List:

- ◆ [:TRIGger:RUNT:SOURce](#)
- ◆ [:TRIGger:RUNT:POLarity](#)
- ◆ [:TRIGger:RUNT:WHEN](#)
- ◆ [:TRIGger:RUNT:WUPPer](#)
- ◆ [:TRIGger:RUNT:WLOWer](#)
- ◆ [:TRIGger:RUNT:ALEVel](#)
- ◆ [:TRIGger:RUNT:BLEVel](#)

:TRIGger:RUNT:SOURce

Syntax :TRIGger:RUNT:SOURce <source>
:TRIGger:RUNT:SOURce?

Description Set or query the trigger source in runt trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:RUNT:SOURce CHANnel2

:TRIGger:RUNT:POLarity

Syntax :TRIGger:RUNT:POLarity <polarity>
:TRIGger:RUNT:POLarity?

Description Set or query the pulse polarity in runt trigger.

Parameter	Name	Type	Range	Default
	<polarity>	Discrete	{POSitive NEGative}	POSitive

- Explanation**
- POSitive: the instrument triggers on the positive runt pulse.
 - NEGative: the instrument triggers on the negative runt pulse.

Return Format The query returns POS or NEG.

Example :TRIGger:RUNT:POLarity NEGative

:TRIGger:RUNT:WHEN

Syntax :TRIGger:RUNT:WHEN <when>

:TRIGger:RUNT:WHEN?

Description Set or query the qualifier in runt trigger.

Parameter	Name	Type	Range	Default
	<when>	Discrete	{NONE GREater LESS GLESS}	NONE

- Explanation**
- NONE: do not set the trigger condition in runt trigger.
 - GREater: trigger when the runt pulse width is greater than the lower limit of the pulse width (refer to [:TRIGger:RUNT:WLOWer](#)).
 - LESS: trigger when the runt pulse width is lower than the upper limit of the pulse width (refer to [:TRIGger:RUNT:WUPPer](#)).
 - GLESS: trigger when the runt pulse width is greater than the lower limit (refer to [:TRIGger:RUNT:WLOWer](#)) and lower than the upper limit of the pulse width (refer to [:TRIGger:RUNT:WUPPer](#)). Note: the lower limit of the pulse width must be lower than the upper limit.

Return Format The query returns NONE, GRE, LESS or GLES.

Example :TRIGger:RUNT:WHEN LESS

Related Commands [:TRIGger:RUNT:WUPPer](#)
[:TRIGger:RUNT:WLOWer](#)

:TRIGger:RUNT:WUPPer

Syntax :TRIGger:RUNT:WUPPer <NR3>

:TRIGger:RUNT:WUPPer?

Description Set or query the pulse width upper limit in runt trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<NR3>	Discrete	16ns to 10s	2μs

Explanation This command is only available when the qualifier (refer to [:TRIGger:RUNT:WHEN](#)) is LESS or GLESS.

Return Format The query returns the pulse width upper limit in scientific notation.

Example :TRIGger:RUNT:WUPPer 0.02 /*Set the pulse width upper limit to 20ms*/
 :TRIGger:RUNT:WUPPer? /*The query returns 2.000000e-02*/

Related Commands [:TRIGger:RUNT:WHEN](#)
[:TRIGger:RUNT:WLOWer](#)

:TRIGger:RUNT:WLOWer

Syntax :TRIGger:RUNT:WLOWer <NR3>

:TRIGger:RUNT:WLOWer?

Description Set or query the pulse width lower limit in runt trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<NR3>	Discrete	8ns to 9.99s	1μs

Explanation This command is only available when the qualifier (refer to [:TRIGger:RUNT:WHEN](#)) is GREater or GLESSs.

Return The query returns the pulse width lower limit in scientific notation.

Format

Example :TRIGger:RUNT:WLOWer 0.01 /*Set the pulse width lower limit to 1ms*/

:TRIGger:RUNT:WLOWer? /*The query returns 1.000000e-03*/

Related [:TRIGger:RUNT:WUPPer](#)
Commands [:TRIGger:RUNT:WHEN](#)

:TRIGger:RUNT:ALEVEL

Syntax :TRIGger:RUNT:ALEVEL <level>

:TRIGger:RUNT:ALEVEL?

Description Set or query the trigger level upper limit in runt trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<level>	Real	(± 5 × VerticalScale from the screen center) - OFFSET	2V

Explanation For VerticalScale, refer to [:CHANnel<n>:SCALE](#). For OFFSET, refer to [:CHANnel<n>:OFFSet](#).

Return The query returns the trigger level upper limit in scientific notation.

Format

Example :TRIGger:RUNT:ALEVEL 0.16 /*Set the trigger level upper limit to 160mV*/

:TRIGger:RUNT:ALEVEL? /*The query returns 1.600000e-01*/

Related [:CHANnel<n>:SCALE](#)
Commands [:CHANnel<n>:OFFSet](#)

:TRIGger:RUNT:BLEVel

Syntax :TRIGger:RUNT:BLEVel <level>

:TRIGger:RUNT:BLEVel?

Description Set or query the trigger level lower limit in runt trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<level>	Real	($\pm 5 \times \text{VerticalScale}$ from the screen center) - OFFSet	0

Explanation For VerticalScale, refer to [:CHANnel<n>:SCALE](#). For OFFSet, refer to [:CHANnel<n>:OFFSet](#).

Return Format The query returns the trigger level lower limit in scientific notation.

Example :TRIGger:RUNT:BLEVel 0.16 /*Set the trigger level lower limit to 160mV*/
:TRIGger:RUNT:BLEVel? /*The query returns 1.600000e-01*/

Related Commands [:CHANnel<n>:SCALE](#)
[:CHANnel<n>:OFFSet](#)

:TRIGger:WINDows (Option)

Command List:

- ◆ [:TRIGger:WINDows:SOURce](#)
- ◆ [:TRIGger:WINDows:SLOPe](#)
- ◆ [:TRIGger:WINDows:POSition](#)
- ◆ [:TRIGger:WINDows:TIME](#)
- ◆ [:TRIGger:WINDows:ALEVEL](#)
- ◆ [:TRIGger:WINDows:BLEVEL](#)

:TRIGger:WINDows:SOURce

Syntax :TRIGger:WINDows:SOURce <source>
:TRIGger:WINDows:SOURce?

Description Set or query the trigger source in windows trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:WINDows:SOURce CHANnel2

:TRIGger:WINDows:SLOPe

Syntax :TRIGger:WINDows:SLOPe <type>
:TRIGger:WINDows:SLOPe?

Description Set or query the windows type in windows trigger.

Parameter	Name	Type	Range	Default
	<type>	Discrete	{POSitive NEGative RFALI}	POSitive

- Explanation**
- POSitive: trigger on the rising edge of the input signal when the voltage level is greater than the preset high trigger level.
 - NEGative: trigger on the falling edge of the input signal when the voltage level is lower than the preset low trigger level.
 - RFALI: trigger on any edge of the input signal when the voltage level meets the preset trigger level.

Return Format The query returns POS, NEG or RFAL.

Example :TRIGger:WINDows:SLOPe NEGative

:TRIGger:WINDows:POSition

Syntax :TRIGger:WINDows:POSition <pos>
:TRIGger:WINDows:POSition?

Description Set or query the trigger position in windows trigger.

Parameter	Name	Type	Range	Default
	<type>	Discrete	{EXIT ENTER TIme}	ENTER

- Explanation**
- EXIT: trigger when the input signal exits the specified trigger level range.
 - ENTER: trigger when the trigger signal enters the specified trigger level range.
 - TIme: used to specify the hold time of the input signal after it enters the specified trigger level range. The instrument triggers when the accumulated hold time equals the windows time.

Return Format The query returns EXIT, ENTER or TIM.

Example :TRIGger:WINDows:POSition ENTER

:TRIGger:WINDows:TIme

Syntax :TRIGger:WINDows:TIme <NR3>
:TRIGger:WINDows:TIme?

Description Set or query the hold time in windows trigger.

Parameter	Name	Type	Range	Default
	<NR3>	Real	8ns to 10s	1μs

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

Example :TRIGger:WINDows:TIme 0.002 /*Set the hold time to 2ms*/
:TRIGger:WINDows:TIme? /*The query returns 2.000000e-03*/

Related Command [:TRIGger:WINDows:POSition](#)

:TRIGger:WINDows:ALEVel

Syntax :TRIGger:WINDows:ALEVel <level>

:TRIGger:WINDows:ALEVel?

Description Set or query the trigger level upper limit in windows trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<level>	Real	($\pm 5 \times \text{VerticalScale}$ from the screen center) - OFFSet	2V

Explanation For VerticalScale, refer to [:CHANnel<n>:SCALE](#). For OFFSet, refer to [:CHANnel<n>:OFFSet](#).

Return Format The query returns the trigger level upper limit in scientific notation.

Example :TRIGger:WINDows:ALEVel 0.16 /*Set the trigger level upper limit to 160mV*/
:TRIGger:WINDows:ALEVel? /*The query returns 1.600000e-01*/

Related Commands [:CHANnel<n>:SCALE](#)
[:CHANnel<n>:OFFSet](#)

:TRIGger:WINDows:BLEVel

Syntax :TRIGger:WINDows:BLEVel <level>

:TRIGger:WINDows:BLEVel?

Description Set or query the trigger level lower limit in windows trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<level>	Real	($\pm 5 \times \text{VerticalScale}$ from the screen center) - OFFSet	0

Explanation For VerticalScale, refer to [:CHANnel<n>:SCALE](#). For OFFSet, refer to [:CHANnel<n>:OFFSet](#).

Return Format The query returns the trigger level lower limit in scientific notation.

Example :TRIGger:WINDows:BLEVel 0.05

Related Commands [:CHANnel<n>:SCALE](#)
[:CHANnel<n>:OFFSet](#)

:TRIGger:DElay (Option)

Command List:

- ◆ [:TRIGger:DElay:SA](#)
- ◆ [:TRIGger:DElay:SLOPA](#)
- ◆ [:TRIGger:DElay:SB](#)
- ◆ [:TRIGger:DElay:SLOPB](#)
- ◆ [:TRIGger:DElay:TYPe](#)
- ◆ [:TRIGger:DElay:TUPPer](#)
- ◆ [:TRIGger:DElay:TLOWer](#)

:TRIGger:DElay:SA

Syntax :TRIGger:DElay:SA <Source>
:TRIGger:DElay:SA?

Description Set or query the trigger source A in delay trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:DElay:SA CHANnel2

:TRIGger:DElay:SLOPA

Syntax :TRIGger:DElay:SLOPA <slope>
:TRIGger:DElay:SLOPA?

Description Set or query the edge type of the trigger source A in delay trigger.

Parameter	Name	Type	Range	Default
	<slope>	Discrete	{POSitive NEGative}	POSitive

Return Format The query returns POS or NEG.

Example :TRIGger:DElay:SLOPA NEGative

:TRIGger:DElay:SB

Syntax :TRIGger:DElay:SB <source>
:TRIGger:DElay:SB?

Description Set or query the trigger source B in delay trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel2

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:DElay:SB CHANnel2

:TRIGger:DElay:SLOPB**Syntax** :TRIGger:DElay:SLOPB <slope>

:TRIGger:DElay:SLOPB?

Description Set or query the edge type of the trigger source B in delay trigger.

Parameter	Name	Type	Range	Default
	<slope>	Discrete	{POSitive NEGative}	POSitive

Return Format The query returns POS or NEG.**Example** :TRIGger:DElay:SLOPB NEGative**:TRIGger:DElay:TYPe****Syntax** :TRIGger:DElay:TYPe <type>

:TRIGger:DElay:TYPe?

Description Set or query the delay type in delay trigger.

Parameter	Name	Type	Range	Default
	<type>	Discrete	{GREater LESS GLEsS GOUT}	GREater

- Explanation**
- GREater: trigger when the time difference (ΔT) between the specified edges of trigger source A and trigger source B is greater than the preset time limit.
 - LESS: trigger when the time difference (ΔT) between the specified edges of trigger source A and trigger source B is lower than the preset time limit.
 - GLEsS: trigger when the time difference (ΔT) between the specified edges of trigger source A and trigger source B is greater than the lower limit of the preset time and lower than the upper limit of the preset time. Note that the time lower limit must be lower than the time upper limit.
 - GOUT: trigger when the time difference (ΔT) between the specified edges of trigger source A and trigger source B is lower than the lower limit of the preset time or greater than the upper limit of the preset time. Note that the time lower limit must be lower than the time upper limit.

Return Format The query returns GOUT, GRE, LESS or GLEsS.**Example** :TRIGger:DElay:TYPe GOUT**Related Commands** [:TRIGger:DElay:TUPPer](#)
[:TRIGger:DElay:TLOWer](#)

:TRIGger:DElay:TUPPer

Syntax :TRIGger:DElay:TUPPer <NR3>
:TRIGger:DElay:TUPPer?

Description Set or query the upper limit of the delay time in delay trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<NR3>	Real	16ns to 10s	2μs

Explanation This command is only available when the delay type (refer to [:TRIGger:DElay:TYPe](#)) is LESS, GOUT or GLESSs.

Return Format The query returns the upper limit of the delay time in scientific notation.

Example :TRIGger:DElay:TUPPer 0.002 /*Set the upper limit of the delay time to 2ms*/
:TRIGger:DElay:TUPPer? /*The query returns 2.000000e-03*/

Related Command [:TRIGger:DElay:TYPe](#)

:TRIGger:DElay:TLOWer

Syntax :TRIGger:DElay:TLOWer <NR3>
:TRIGger:DElay:TLOWer?

Description Set or query the lower limit of the delay time in delay trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<NR3>	Real	Related to the delay type: GREater: 8ns to 10s GOUT or GLESSs: 8ns to 9.99s	1μs

Explanation This command is only available when the delay type (refer to [:TRIGger:DElay:TYPe](#)) is GREater, GOUT or GLESSs.

Return Format The query returns the lower limit of the delay time in scientific notation.

Example :TRIGger:DElay:TLOWer 0.002 /*Set the lower limit of the delay time to 2ms*/
:TRIGger:DElay:TLOWer? /*The query returns 2.000000e-03*/

Related Commands [:TRIGger:DElay:TYPe](#)
[:TRIGger:DElay:TUPPer](#)

:TRIGger:SHOLd (Option)

Command List:

- ◆ [:TRIGger:SHOLd:DSrc](#)
- ◆ [:TRIGger:SHOLd:Csrc](#)
- ◆ [:TRIGger:SHOLd:SLOPe](#)
- ◆ [:TRIGger:SHOLd:PATtern](#)
- ◆ [:TRIGger:SHOLd:TYPe](#)
- ◆ [:TRIGger:SHOLd:STIMe](#)
- ◆ [:TRIGger:SHOLd:HTIMe](#)

:TRIGger:SHOLd:DSrc

Syntax :TRIGger:SHOLd:DSrc <source>
:TRIGger:SHOLd:DSrc?

Description Set or query the data source in setup/hold trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:SHOLd:DSrc CHANnel1

:TRIGger:SHOLd:Csrc

Syntax :TRIGger:SHOLd:Csrc <source>
:TRIGger:SHOLd:Csrc?

Description Set or query the clock source in setup/hold trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel2

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:SHOLd:Csrc CHANnel2

:TRIGger:SHOLd:SLOPe

Syntax :TRIGger:SHOLd:SLOPe <slope>
:TRIGger:SHOLd:SLOPe?

Description Set or query the edge type in setup/hold trigger.

Parameter	Name	Type	Range	Default
	<slope>	Discrete	{POSitive NEGative}	POSitive

Return Format The query returns POS or NEG.

Example :TRIGger:SHOLd:SLOPe NEGative

:TRIGger:SHOLd:PATtern

Syntax :TRIGger:SHOLd:PATtern <pattern>
:TRIGger:SHOLd:PATtern?

Description Set or query the data type in setup/hold trigger.

Parameter	Name	Type	Range	Default
	<pattern>	Discrete	{H L}	H

Explanation H: high level
L: low level

Return Format The query returns H or L.

Example :TRIGger:SHOLd:PATtern L

:TRIGger:SHOLd:TYPe

Syntax :TRIGger:SHOLd:TYPe <type>
:TRIGger:SHOLd:TYPe?

Description Set or query the setup type in setup/hold trigger.

Parameter	Name	Type	Range	Default
	<type>	Discrete	{SETup HOLd SETHOLd}	SETup

Explanation

- SETup: set the time that the data stays stable and constant before the clock edge appears ([:TRIGger:SHOLd:STIME](#)).
- HOLd: set the time that the data stays stable and constant after the clock edge appears ([:TRIGger:SHOLd:HTIME](#)).
- SETHOLd: set the time that the data stays stable and constant before and after the clock edge appears ([:TRIGger:SHOLd:STIME](#) and [:TRIGger:SHOLd:HTIME](#)).

Return Format The query returns SET, HOL or SETHOL.

Example :TRIGger:SHOLd:TYPe SETHOLd

Related Commands [:TRIGger:SHOLd:HTIME](#)
[:TRIGger:SHOLd:STIME](#)

:TRIGger:SHOLd:STIME**Syntax** :TRIGger:SHOLd:STIME <NR3>

:TRIGger:SHOLd:STIME?

Description Set or query the setup time in setup/hold trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<NR3>	Real	8ns to 1s	1 μ s

Explanation This command is only available when the setup type (refer to [:TRIGger:SHOLd:TYPE](#)) is SETup or SETHOLd.**Return Format** The query returns the setup time in scientific notation.**Example** :TRIGger:SHOLd:STIME 0.002 /*Set the setup time to 2ms*/
:TRIGger:SHOLd:STIME? /*The query returns 2.000000e-03*/**Related Commands** [:TRIGger:SHOLd:TYPE](#)
[:TRIGger:SHOLd:HTIME](#)**:TRIGger:SHOLd:HTIME****Syntax** :TRIGger:SHOLd:HTIME <NR3>

:TRIGger:SHOLd:HTIME?

Description Set or query the hold time in setup/hold trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<NR3>	Real	8ns to 1s	1 μ s

Explanation This command is only available when the setup type (refer to [:TRIGger:SHOLd:TYPE](#)) is HOLd or SETHOLd.**Return Format** The query returns the hold time in scientific notation.**Example** :TRIGger:SHOLd:HTIME 0.002 /*Set the hold time to 2ms*/
:TRIGger:SHOLd:HTIME? /*The query returns 2.000000e-03*/**Related Commands** [:TRIGger:SHOLd:TYPE](#)
[:TRIGger:SHOLd:STIME](#)

:TRIGger:NEDGe (Option)

Command List:

- ◆ [:TRIGger:NEDGe:SOURce](#)
- ◆ [:TRIGger:NEDGe:SLOPe](#)
- ◆ [:TRIGger:NEDGe:IDLE](#)
- ◆ [:TRIGger:NEDGe:EDGE](#)
- ◆ [:TRIGger:NEDGe:LEVel](#)

:TRIGger:NEDGe:SOURce

Syntax :TRIGger:NEDGe:SOURce <source>
:TRIGger:NEDGe:SOURce?

Description Set or query the trigger source in Nth edge trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:NEDGe:SOURce CHANnel2

:TRIGger:NEDGe:SLOPe

Syntax :TRIGger:NEDGe:SLOPe <slope>
:TRIGger:NEDGe:SLOPe?

Description Set or query the edge type in Nth edge trigger.

Parameter	Name	Type	Range	Default
	<slope>	Discrete	{POSitive NEGative}	POSitive

- Explanation**
- POSitive: trigger on the rising edge of the input signal when the voltage level meets the specified trigger level.
 - NEGative: trigger on the falling edge of the input signal when the voltage level meets the specified trigger level.

Return Format The query returns POS or NEG.

Example :TRIGger:NEDGe:SLOPe NEGative

:TRIGger:NEDGE:IDLE

Syntax :TRIGger:NEDGE:IDLE <NR3>

:TRIGger:NEDGE:IDLE?

Description Set or query the idle time in Nth edge trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<NR3>	Real	16ns to 10s	1μs

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

Example :TRIGger:NEDGE:IDLE 0.002 /*Set the idle time to 2ms*/
:TRIGger:NEDGE:IDLE? /*The query returns 2.000000e-03*/

:TRIGger:NEDGE:EDGE

Syntax :TRIGger:NEDGE:EDGE <NR1>

:TRIGger:NEDGE:EDGE?

Description Set or query the edge number in Nth edge trigger.

Parameter	Name	Type	Range	Default
	<NR1>	Integer	1 to 65535	2

Return Format The query returns an integer between 1 and 65535.

Example :TRIGger:NEDGE:EDGE 20

:TRIGger:NEDGE:LEVEL

Syntax :TRIGger:NEDGE:LEVEL <level>

:TRIGger:NEDGE:LEVEL?

Description Set or query the trigger level in Nth edge trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<level>	Real	(± 5 × VerticalScale from the screen center) - OFFSet	0

Explanation For VerticalScale, refer to [:CHANnel<n>:SCALE](#). For OFFSet, refer to [:CHANnel<n>:OFFSet](#).

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

Example :TRIGger:NEDGE:LEVEL 0.16 /*Set the the trigger level to 160mV*/
:TRIGger:NEDGE:LEVEL? /*The query returns 1.600000e-01*/

Related Commands [:CHANnel<n>:SCALE](#)
[:CHANnel<n>:OFFSet](#)

:TRIGger:RS232 (Option)

Command List:

- ◆ [:TRIGger:RS232:SOURce](#)
- ◆ [:TRIGger:RS232:WHEN](#)
- ◆ [:TRIGger:RS232:PARity](#)
- ◆ [:TRIGger:RS232:STOP](#)
- ◆ [:TRIGger:RS232:DATA](#)
- ◆ [:TRIGger:RS232:WIDTh](#)
- ◆ [:TRIGger:RS232:BAUD](#)
- ◆ [:TRIGger:RS232:BUSeR](#)
- ◆ [:TRIGger:RS232:LEVel](#)

:TRIGger:RS232:SOURce

Syntax :TRIGger:RS232:SOURce <source>
:TRIGger:RS232:SOURce?

Description Set or query the trigger source in RS232 trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:RS232:SOURce CHANnel2

:TRIGger:RS232:WHEN

Syntax :TRIGger:RS232:WHEN <when>
:TRIGger:RS232:WHEN?

Description Set or query the trigger condition in RS232 trigger.

Parameter	Name	Type	Range	Default
	<when>	Discrete	{START ERRor PARity DATA}	START

- Explanation**
- START: trigger on the start frame position.
 - ERRor: trigger when error frame is detected.
 - PARity: trigger when check error is detected.
 - DATA: trigger on the last bit of the preset data bits and even-odd check bits.

Return Format The query returns STAR, ERR, PAR or DATA.

Example :TRIGger:RS232:WHEN ERRor

Related Command [:TRIGger:RS232:DATA](#)

:TRIGger:RS232:PARity

Syntax :TRIGger:RS232:PARity <parity>

:TRIGger:RS232:PARity?

Description Set or query the parity type when the trigger condition is ERRor or PARity in RS232 trigger.

Parameter	Name	Type	Range	Default
	<parity>	Discrete	{EVEN ODD NONE}	NONE

Explanation When the trigger condition is PARity, the parity type cannot be set to NONE. At this point, the default parity type is ODD.

Return Format The query returns EVEN, ODD or NONE.

Example :TRIGger:RS232:PARity EVEN

Related Command [:TRIGger:RS232:WHEN](#)

:TRIGger:RS232:STOP

Syntax :TRIGger:RS232:STOP <bit>

:TRIGger:RS232:STOP?

Description Set or query the stop bit when the trigger condition is ERRor in RS232 trigger.

Parameter	Name	Type	Range	Default
	<bit>	Discrete	{1 2}	1

Return Format The query returns 1 or 2.

Example :TRIGger:RS232:STOP 2

Related Command [:TRIGger:RS232:WHEN](#)

:TRIGger:RS232:DATA

Syntax :TRIGger:RS232:DATA <data>

:TRIGger:RS232:DATA?

Description Set or query the data when the trigger condition is DATA in RS232 trigger.

Parameter	Name	Type	Range	Default
	<data>	Integer	0 to 2^n-1	90

Explanation In 2^n-1 , n is the current data bits and can be 5, 6, 7 or 8.

Return Format The query returns an integer.

Example :TRIGger:RS232:DATA 10

Related Commands [:TRIGger:RS232:WIDTH](#)
[:TRIGger:RS232:WHEN](#)

:TRIGger:RS232:WIDTH

Syntax :TRIGger:RS232:WIDTH <width>

:TRIGger:RS232:WIDTH?

Description Set or query the data bits when the trigger condition is DATA in RS232 trigger.

Parameter	Name	Type	Range	Default
	<width>	Discrete	{5 6 7 8}	8

Return Format The query returns 5, 6, 7 or 8.

Example :TRIGger:RS232:WIDTH 6

Related Command [:TRIGger:RS232:WHEN](#)

:TRIGger:RS232:BAUD

Syntax :TRIGger:RS232:BAUD <baud_rate>

:TRIGger:RS232:BAUD?

Description Set or query the baud rate in RS232 trigger. The default unit is bps.

Parameter	Name	Type	Range	Default
	<baud_rate>	Discrete	{2400 4800 9600 19200 38400 57600 115200 230400 460800 921600 1000000 USER}	2400

Return Format The query returns an integer.

Example :TRIGger:RS232:BAUD 4800

Related Command [:TRIGger:RS232:BUSer](#)

:TRIGger:RS232:BUSer

Syntax :TRIGger:RS232:BUSer <user baud>

:TRIGger:RS232:BUSer?

Description Set or query the user-defined baud rate in RS232 trigger. The default unit is bps.

Parameter	Name	Type	Range	Default
	<user baud>	Integer	1 to 900000	9600

Return Format The query returns an integer.

Example :TRIGger:RS232:BUSer 50000

Related Command [:TRIGger:RS232:BAUD](#)

:TRIGger:RS232:LEVel

Syntax :TRIGger:RS232:LEVel <level>

:TRIGger:RS232:LEVel?

Description Set or query the trigger level in RS232 trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<level>	Real	($\pm 5 \times \text{VerticalScale}$ from the screen center) - OFFSet	0

Explanation For VerticalScale, refer to [:CHANnel<n>:SCALE](#). For OFFSet, refer to [:CHANnel<n>:OFFSet](#).

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

Example :TRIGger:RS232:LEVel 0.16 /*Set the trigger level to 160mV*/

:TRIGger:RS232:LEVel? /*The query returns 1.600000e-01*/

Related [:CHANnel<n>:SCALE](#)

Commands [:CHANnel<n>:OFFSet](#)

:TRIGger:IIC (Option)

Command List:

- ◆ [:TRIGger:IIC:SCL](#)
- ◆ [:TRIGger:IIC:SDA](#)
- ◆ [:TRIGger:IIC:WHEN](#)
- ◆ [:TRIGger:IIC:AWIDth](#)
- ◆ [:TRIGger:IIC:ADDRes](#)
- ◆ [:TRIGger:IIC:DIRection](#)
- ◆ [:TRIGger:IIC:DATA](#)
- ◆ [:TRIGger:IIC:CLEVel](#)
- ◆ [:TRIGger:IIC:DLEVel](#)

:TRIGger:IIC:SCL

Syntax :TRIGger:IIC:SCL <source>
:TRIGger:IIC:SCL?

Description Set or query the channel source of SCL in I2C trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:IIC:SCL CHANnel2

:TRIGger:IIC:SDA

Syntax :TRIGger:IIC:SDA <source>
:TRIGger:IIC:SDA?

Description Set or query the channel source of SDA in I2C trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel2

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:IIC:SDA CHANnel2

:TRIGger:IIC:WHEN

Syntax :TRIGger:IIC:WHEN <trig_type>

:TRIGger:IIC:WHEN?

Description Set or query the trigger condition in I2C trigger.

Parameter	Name	Type	Range	Default
	<trig_type>	Discrete	{START REStart STOP NACKnowledge ADDRess DATA ADATa}	START

- Explanation**
- START: trigger when the SDA data transitions from high to low while the SCL is high.
 - REStart: trigger when another start condition occurs before a stop condition.
 - STOP: trigger when the SDA data transitions from low to high while the SCL is high.
 - NACKnowledge: trigger when the SDA data is high during any acknowledgement of the SCL clock position.
 - ADDRess: trigger on the clock (SCL) edge corresponding to the byte of data (SDA) behind the preset address (Write, Read or R/W direction).
 - DATA: the trigger searches for the control byte value on the data line (SDA) following which there is a reading bit and an acknowledgement bit and then searches for the specified data value. When this event occurs, the oscilloscope will trigger on the clock edge of the acknowledgement bit behind the data byte.
 - ADATa: trigger when the "Address" and "Data" conditions are met at the same time.

Return Format The query returns STAR, STOP, NACK, REST, ADDR, DATA or ADAT.

Example :TRIGger:IIC:WHEN REStart

:TRIGger:IIC:AWIDth

Syntax :TRIGger:IIC:AWIDth <bits>

:TRIGger:IIC:AWIDth?

Description Set or query the address bits when the trigger condition is ADDRess or ADATa in I2C trigger.

Parameter	Name	Type	Range	Default
	<bits>	Discrete	{7 8 10}	7

Return Format The query returns 7, 8 or 10.

Example :TRIGger:IIC:AWIDth 10

Related Commands [:TRIGger:IIC:DIRection](#)
[:TRIGger:IIC:ADDRess](#)

:TRIGger:IIC:ADDRESS

Syntax :TRIGger:IIC:ADDRESS <adr>

:TRIGger:IIC:ADDRESS?

Description Set or query the address when the trigger condition is ADDRESS or ADATA in I2C trigger.

Parameter	Name	Type	Range	Default
	<adr>	Integer	0 to $2^n - 1$: 0 to 127, 0 to 255 or 0 to 1023	1

Explanation In $2^n - 1$, n is the current address bits.

Return Format The query returns an integer.

Example :TRIGger:IIC:ADDRESS 100

Related Command [:TRIGger:IIC:AWIDTH](#)

:TRIGger:IIC:DIRection

Syntax :TRIGger:IIC:DIRection <dir>

:TRIGger:IIC:DIRection?

Description Set or query the data direction when the trigger condition is ADDRESS or ADATA in I2C trigger.

Parameter	Name	Type	Range	Default
	<dir>	Discrete	{READ WRITe RWRite}	READ

Explanation This command is invalid when the address bits is 8.

Return Format The query returns READ, WRIT or RWR.

Example :TRIGger:IIC:DIRection RWRite

Related Command [:TRIGger:IIC:AWIDTH](#)

:TRIGger:IIC:DATA

Syntax :TRIGger:IIC:DATA <data>

:TRIGger:IIC:DATA?

Description Set or query the data when the trigger condition is DATA or ADATA in I2C trigger.

Parameter	Name	Type	Range	Default
	<data>	Integer	0 to $2^{40} - 1$	82

Explanation The range of <data> is related to the byte length. The maximum byte length is 5, namely 40 bits binary data. Thus, the range of <data> is from 0 to $2^{40} - 1$.

Return Format The query returns an integer.

Example :TRIGger:IIC:DATA 64

Related Command [:TRIGger:IIC:WHEN](#)

:TRIGger:IIC:CLeVel

Syntax :TRIGger:IIC:CLeVel <level>

:TRIGger:IIC:CLeVel?

Description Set or query the trigger level of SCL in I2C trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<level>	Real	($\pm 5 \times \text{VerticalScale}$ from the screen center) - OFFSet	0

Explanation For VerticalScale, refer to [:CHANnel<n>:SCALe](#). For OFFSet, refer to [:CHANnel<n>:OFFSet](#).

Return Format The query returns the trigger level of SCL in scientific notation.

Example :TRIGger:IIC:CLeVel 0.16 /*Set the trigger level of SCL to 160mV*/
:TRIGger:IIC:CLeVel? /*The query returns 1.600000e-01*/

Related Commands [:CHANnel<n>:SCALe](#)
[:CHANnel<n>:OFFSet](#)

:TRIGger:IIC:DLeVel

Syntax :TRIGger:IIC:DLeVel <level>

:TRIGger:IIC:DLeVel?

Description Set or query the trigger level of SDA in I2C trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<level>	Real	($\pm 5 \times \text{VerticalScale}$ from the screen center) - OFFSet	0

Explanation For VerticalScale, refer to [:CHANnel<n>:SCALe](#). For OFFSet, refer to [:CHANnel<n>:OFFSet](#).

Return Format The query returns the trigger level of SDA in scientific notation.

Example :TRIGger:IIC:DLeVel 0.16 /*Set the trigger level of SDA to 160mV*/
:TRIGger:IIC:DLeVel? /*The query returns 1.600000e-01*/

Related Commands [:CHANnel<n>:SCALe](#)
[:CHANnel<n>:OFFSet](#)

:TRIGger:SPI (Option)

Command List:

- ◆ [:TRIGger:SPI:SCL](#)
- ◆ [:TRIGger:SPI:SDA](#)
- ◆ [:TRIGger:SPI:WHEN](#)
- ◆ [:TRIGger:SPI:WIDTh](#)
- ◆ [:TRIGger:SPI:DATA](#)
- ◆ [:TRIGger:SPI:TIMeout](#)
- ◆ [:TRIGger:SPI:SLOPe](#)
- ◆ [:TRIGger:SPI:CLEVel](#)
- ◆ [:TRIGger:SPI:DLEVel](#)

:TRIGger:SPI:SCL

Syntax :TRIGger:SPI:SCL <source>
:TRIGger:SPI:SCL?

Description Set or query the channel source of SCL in SPI trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:SPI:SCL CHANnel1 /*Set the channel source of SCL to CH1*/

:TRIGger:SPI:SDA

Syntax :TRIGger:SPI:SDA <source>
:TRIGger:SPI:SDA?

Description Set or query the channel source of SDA in SPI trigger.

Parameter	Name	Type	Range	Default
	<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel2

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :TRIGger:SPI:SDA CHANnel2 /*Set the channel source of SDA to CH2*/

:TRIGger:SPI:WHEN

Syntax :TRIGger:SPI:WHEN <trig_type>

:TRIGger:SPI:WHEN?

Description Set or query the trigger condition in SPI trigger.

Parameter	Name	Type	Range	Default
	<trig_type>	Discrete	{CS TIMEout}	CS

Explanation When the trigger condition is set to TIMEout, you can use the [:TRIGger:SPI:TIMEout](#) command to set the timeout value.

Return The query returns CS or TIM.

Format

Example :TRIGger:SPI:WHEN TIMEout /*Set the trigger condition to TIMEout*/

Related Command [:TRIGger:SPI:TIMEout](#)

:TRIGger:SPI:WIDTH

Syntax :TRIGger:SPI:WIDTH <width>

:TRIGger:SPI:WIDTH?

Description Set or query the data bits of the SDA channel in SPI trigger.

Parameter	Name	Type	Range	Default
	<width>	Integer	4 to 32	8

Return The query returns an integer.

Format

Example :TRIGger:SPI:WIDTH 10 /*Set the data bits of the SDA channel to 10*/

Related Command [:TRIGger:SPI:DATA](#)

:TRIGger:SPI:DATA

Syntax :TRIGger:SPI:DATA <data>

:TRIGger:SPI:DATA?

Description Set or query the data in SPI trigger.

Parameter	Name	Type	Range	Default
	<data>	Integer	0 to $2^{32}-1$	0

Explanation The range of <data> is related to the data bits. The maximum data bits is 32. Thus, the range of <data> is from 0 to $2^{32}-1$.

Return The query returns an integer.

Format

Example :TRIGger:SPI:DATA 5 /*Set the data to 5*/

Related Command [:TRIGger:SPI:WIDTH](#)

:TRIGger:SPI:TIMEout

Syntax :TRIGger:SPI:TIMEout <time_value>

:TRIGger:SPI:TIMEout?

Description Set or query the timeout value when the trigger condition is TIMEout in SPI trigger. The default unit is s.

Parameter	Name	Type	Range	Default
	<time_value>	Real	100ns to 1s	1μs

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

Example :TRIGger:SPI:TIMEout 0.001 /*Set the timeout value to 1ms*/
:TRIGger:SPI:TIMEout? /*The query returns 1.000000e-03*/

Related Command [:TRIGger:SPI:WHEN](#)

:TRIGger:SPI:SLOPe

Syntax :TRIGger:SPI:SLOPe <slope>

:TRIGger:SPI:SLOPe?

Description Set or query the clock edge in SPI trigger.

Parameter	Name	Type	Range	Default
	<slope>	Discrete	{POSitive NEGative}	POSitive

Explanation POSitive: sample the SDA data on the rising edge of the clock.
NEGative: sample the SDA data on the falling edge of the clock.

Return Format The query returns POS or NEG.

Example :TRIGger:SPI:SLOPe POSitive /*Set the clock edge to POSitive*/

:TRIGger:SPI:CLEVel

Syntax :TRIGger:SPI:CLEVel <level>

:TRIGger:SPI:CLEVel?

Description Set or query the trigger level of the SCL channel in SPI trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<level>	Real	(± 5 × VerticalScale from the screen center) - OFFSet	0

Explanation For VerticalScale, refer to [:CHANnel<n>:SCALE](#). For OFFSet, refer to [:CHANnel<n>:OFFSet](#).

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

Example :TRIGger:SPI:CLEVel 0.16 /*Set the trigger level to 160mV*/
:TRIGger:SPI:CLEVel? /*The query returns 1.600000e-01*/

Related Commands [:CHANnel<n>:SCALE](#)
[:CHANnel<n>:OFFSet](#)

:TRIGger:SPI:DLEVel

Syntax :TRIGger:SPI:DLEVel <level>

:TRIGger:SPI:DLEVel?

Description Set or query the trigger level of the SDA channel in SPI trigger. The unit is the same as the current amplitude unit.

Parameter	Name	Type	Range	Default
	<level>	Real	($\pm 5 \times \text{VerticalScale}$ from the screen center) - OFFSet	0

Explanation For VerticalScale, refer to [:CHANnel<n>:SCALE](#). For OFFSet, refer to [:CHANnel<n>:OFFSet](#).

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

Example :TRIGger:SPI:DLEVel 0.16 /*Set the trigger level to 160mV*/
:TRIGger:SPI:DLEVel? /*The query returns 1.600000e-01*/

Related Commands [:CHANnel<n>:SCALE](#)
[:CHANnel<n>:OFFSet](#)

:WAVEform Commands

The :WAVEform commands are used to read the waveform data and its related settings. :WAVEform:MODE is used to set the reading mode of the waveform data. In different mode, the definition of each parameter is different, as shown in Figure 2-1 and Figure 2-2.

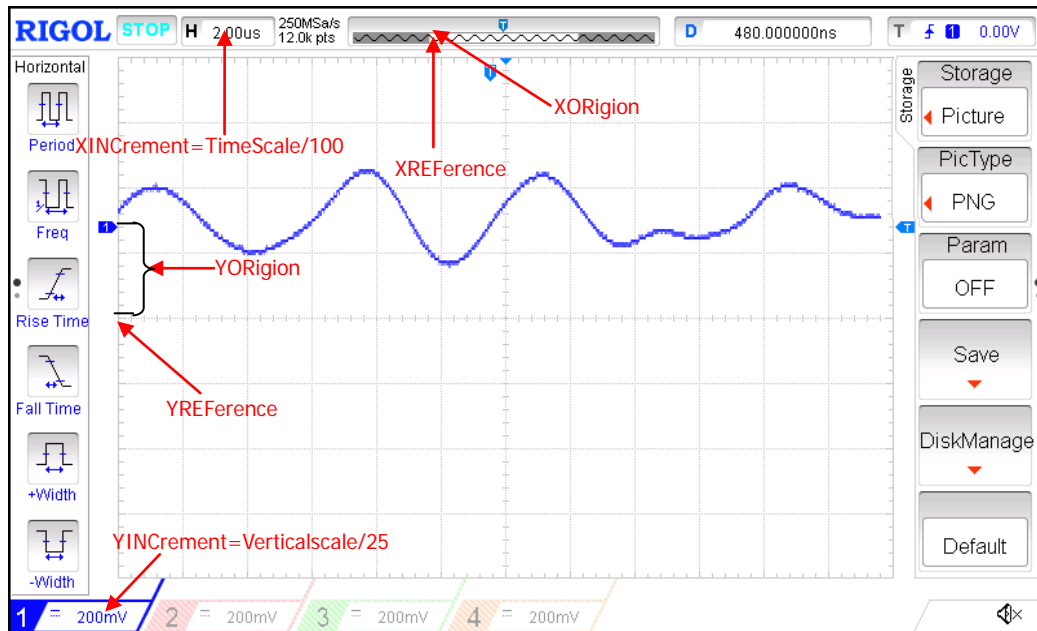


Figure 2-1 Parameter Definitions in NORMAL Mode

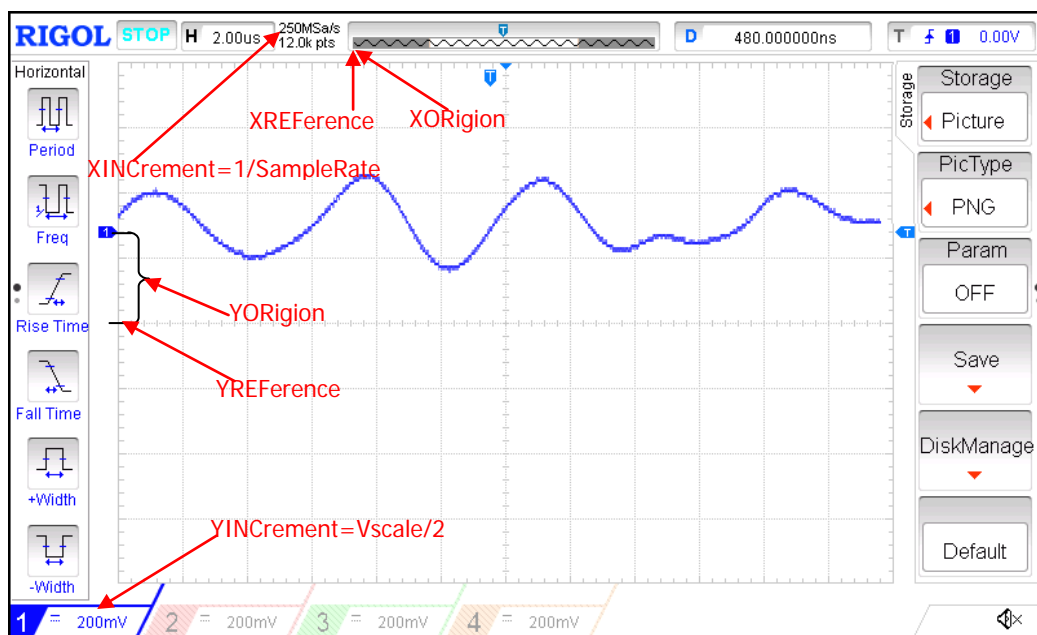


Figure 2-2 Paramter Definitions in RAW Mode

Command List:

- ◆ [:WAVeform:SOURce](#)
- ◆ [:WAVeform:MODE](#)
- ◆ [:WAVeform:FORMat](#)
- ◆ [:WAVeform:DATA?](#)
- ◆ [:WAVeform:XINCrement?](#)
- ◆ [:WAVeform:XORigin?](#)
- ◆ [:WAVeform:XREFerence?](#)
- ◆ [:WAVeform:YINCrement?](#)
- ◆ [:WAVeform:YORigin?](#)
- ◆ [:WAVeform:YREFerence?](#)
- ◆ [:WAVeform:START](#)
- ◆ [:WAVeform:STOP](#)
- ◆ [:WAVeform:PREamble?](#)

:WAVeform:SOURce

Syntax :WAVeform:SOURce <source>

:WAVeform:SOURce?

Description Set or query the channel of which the waveform data will be read.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1 CHANnel2 CHANnel3 CHANnel4 MATH}	CHANnel1

Explanation If the MATH channel is selected, only NORMal can be selected in [:WAVeform:MODE](#).

Return Format The query returns CHAN1, CHAN2, CHAN3 or CHAN4.

Example :WAVeform:SOURce CHANnel2 /*Set the channel to CH2*/

Related Command [:WAVeform:MODE](#)

:WAVeform:MODE

Syntax :WAVeform:MODE <mode>

:WAVeform:MODE?

Description Set or query the reading mode used by [:WAVeform:DATA?](#).

Parameter	Name	Type	Range	Default
	<mode>	Discrete	{NORMal MAXimum RAW}	NORMal

- Explanation**
- NORMal: read the waveform data displayed on the screen.
 - MAXimum: read the waveform data displayed on the screen when the instrument is in the run state and the waveform data in the internal memory in the stop state.
 - RAW: read the waveform data in the internal memory. Note that the waveform data in the internal memory can only be read when the oscilloscope is in the stop state and the oscilloscope can not be operated.
 - If the MATH channel is selected, only the NORMal mode is valid.

Return Format The query returns NORM, MAX or RAW.

Example :WAVeform:MODE RAW /*Set the reading mode to RAW*/

Related Command [:WAVeform:DATA?](#)

:WAVeform:FORMat

Syntax :WAVeform:FORMat <format>

:WAVeform:FORMat?

Description Set or query the return format of the waveform data.

Parameter	Name	Type	Range	Default
	<format>	Discrete	{WORD BYTE ASCii}	BYTE

- Explanation**
- BYTE: a waveform point occupies one byte (namely 8 bits).
 - WORD: a waveform point occupies two bytes (namely 16 bits) in which the lower 8 bits are valid and the higher 8 bits are 0.
 - ASCii: return the waveform points in character number. Waveform points are returned in scientific notation and separated by commas.

Return Format The query returns WORD, BYTE or ASC.

Example :WAVeform:FORMat WORD /*Set the return format to WORD*/

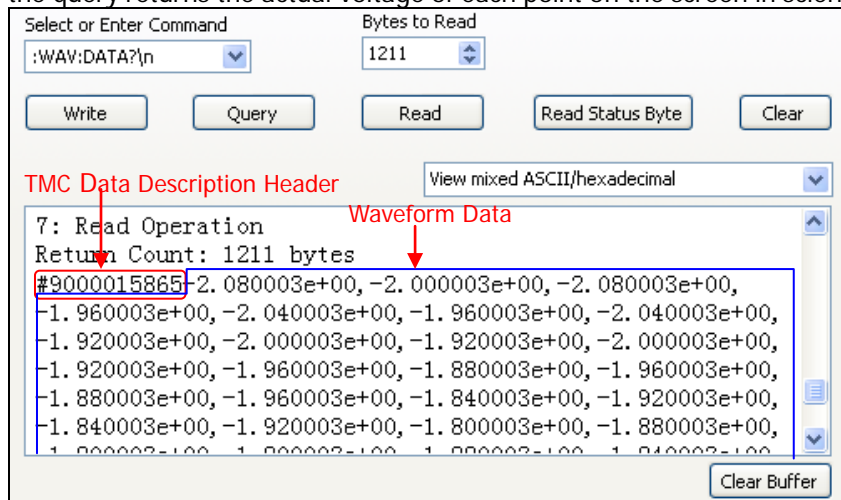
:WAVeform:DATA?

Syntax :WAVeform:DATA?

Description Read the waveform data.

- Explanation**
- Procedures of the screen waveform data reading:
 - 1) Set the channel: :WAV:SOURce CHAN1
 - 2) Select the NORMAl mode: :WAV:MODE NORM
 - 3) Read the screen waveform data: :WAV:DATA?
 - Procedures of the internal memory waveform data reading:
 - 1) Set to the STOP state: :STOP
 - 2) Set the channel: :WAV:SOURce CHAN1
 - 3) Select the RAW mode: :WAV:MODE RAW
 - 4) Read the waveform data in the internal memory: :WAV:DATA?

- Return Format**
- The data returned contains 2 parts: the TMC data description header and the waveform data. The format of the TMC data description header is #90000dddd which is followed the waveform data. Wherein, dddd denotes the number of bytes of the effective waveform data. For example, as shown in the figure below, #9000015865 is the TMC data description header. Wherein, 15865 denotes that there are 15 865 bytes of effective data and it is followed by the waveform data read. The return format of the data is ASCII.
 - When the return format of the waveform data is ASCII (refer to :WAVeform:FORMat), the query returns the actual voltage of each point on the screen in scientific notation.



- When reading the waveform data in the internal memory, the data returned each time may be the data in one area in the internal memory (you can set the start point and stop point using :WAVeform:START and :WAVeform:STOP). Each part of the data returned contains the TMC data description header and the waveform data from two adjacent parts are continuous.

- Related Commands**
- [:WAVeform:SOURce](#)
 - [:WAVeform:MODE](#)
 - [:WAVeform:FORMat](#)
 - [:WAVeform:START](#)
 - [:WAVeform:STOP](#)
 - [:WAVeform:YREFerence?](#)

:WAVeform:XINCrement?

Syntax :WAVeform:XINCrement?

Description Query the time difference between two neighboring points of the specified channel source in the X direction.

Explanation

- The returned value is related to the current data reading mode:
In the NORMAl mode, XINCrement = TimeScale/100;
In the RAW mode, XINCrement = 1/SampleRate
- The unit is related to the current channel source:
when the channel source is one from CHANnel1 to CHANnel4, the unit is s;
when the channel source is MATH and the operation type is FFT, the unit is Hz.

Return Format The query returns the XINCrement in scientific notation.

Example :WAVeform:XINCrement? /*The query returns 1.000000e-08*/

Related Command [:WAVeform:SOURce](#)

:WAVeform:XORigin?

Syntax :WAVeform:XORigin?

Description Query the time from the trigger point to the reference time of the specified channel source in the X direction.

Explanation The unit is related to the current channel source:
when the channel source is one from CHANnel1 to CHANnel4, the unit is s;
when the channel source is MATH and the operation type is FFT, the unit is Hz.

Return Format The query returns the XORigin in scientific notation.

Example :WAVeform:XORigin? /*The query returns -6.000000e-06*/

Related Commands [:WAVeform:SOURce](#)
[:WAVeform:XREFerence?](#)

:WAVeform:XREFerence?

Syntax :WAVeform:XREFerence?

Description Query the reference time of the specified channel source in the X direction

Return Format The query returns 0 (namely the first point on the screen or in the internal memory).

Related Command [:WAVeform:SOURce](#)

:WAVeform:YINCrement?

Syntax :WAVeform:YINCrement?

Description Query the waveform increment of the specified channel source in the Y direction. The unit is the same as the current amplitude unit.

Explanation $YINCrement = VerticalScale / 25$

Return Format The query returns the YINCrement in scientific notation.

Example :WAVeform:YINCrement? /*The query returns 8.000000e-03*/

Related Command [:WAVeform:SOURce](#)

:WAVeform:YORigin?

Syntax :WAVeform:YORigin?

Description Query the vertical offset relative to the vertical reference position of the specified channel source in the Y direction.

Explanation $YORigin = VerticalOffset / YINCrement$.

Return Format The query returns an integer.

Example :WAVeform:YORigin? /*The query returns 0*/

Related Commands [:WAVeform:SOURce](#)
[:WAVeform:YREFerence?](#)

:WAVeform:YREFerence?

Syntax :WAVeform:YREFerence?

Description Query the vertical reference position of the specified channel source in the Y direction.

Explanation The YREFerence is fixed at 127 (the bottom of the screen is 0 and the top is 255).

Return Format The query returns an integer.

Related Command [:WAVeform:SOURce](#)

:WAVEform:START

Syntax :WAVEform:START <sta>

:WAVEform:START?

Description Set or query the start position of internal memory waveform reading.

Parameter	Name	Type	Range	Default
	<sta>	Integer	NORMAL: 1 to 1200 MAX: 1 to the number of effective points currently on the screen RAW: 1 to the current maximum memory depth	1

Return Format The query returns an integer.

Example :WAVEform:START 100 /*Set the start position to 100*/
:WAVEform:START? /*The query returns 100*/

Related Commands [:ACQUIRE:MDEPTH](#)
[:WAVEform:MODE](#)

:WAVEform:STOP

Syntax :WAVEform:STOP <stop>

:WAVEform:STOP?

Description Set or query the stop position of internal memory waveform reading.

Parameter	Name	Type	Range	Default
	<stop>	Integer	NORMAL: 1 to 1200 MAX: 1 to the number of effective point currently on the screen RAW: 1 to the current maximum memory depth	1200

Return Format The query returns an integer.

Example :WAVEform:STOP 500 /*Set the stop position to 500*/
:WAVEform:STOP? /*The query returns 500*/

Related Commands [:ACQUIRE:MDEPTH](#)
[:WAVEform:MODE](#)

:WAVeform:PREamble?

Syntax :WAVeform:PREamble?

Description Query and return all the waveform parameters.

Return The query returns 10 waveform parameters separated by ",":

Format <format>,<type>,<points>,<count>,<xincrement>,<xorigin>,<xreference>,<yincrement>,<yorigin>,<yreference>

Wherein,

<format>: 0 (BYTE), 1 (WORD) or 2 (ASC).

<type>: 0 (NORMal), 1 (MAXimum) or 2 (RAW).

<points>: an integer between 1 and 12000000.

<count>: the number of averages in the average sample mode and 1 in other modes.

<xincrement>: the time difference between two neighboring points in the X direction.

<xorigin>: the time from the trigger point to the "Reference Time" in the X direction.

<xreference>: the reference time of the data point in the X direction.

<yincrement>: the waveform increment in the Y direction.

<yorigin>: the vertical offset relative to the "Vertical Reference Position" in the Y direction.

<yreference>: the vertical reference position in the Y direction.

Example :WAVeform:PREamble? /*The query returns
0,2,600,2,0.000000,-0.000006,0,0.008000,50,127*/

Related [:WAVeform:FORMat](#)
Commands [:WAVeform:MODE](#)

[:ACQuire:AVERages](#)

[:WAVeform:XINCrement?](#)

[:WAVeform:XREFerence?](#)

[:WAVeform:YINCrement?](#)

[:WAVeform:YORigin?](#)

[:WAVeform:YREFerence?](#)

Chapter 3 Programming Demos

This chapter lists some programming demos to illustrate how to use commands to realize the common functions of the oscilloscope in the development environments of Excel, LabVIEW, Matlab, Visual Basic 6.0 and Visual C++ 6.0. All the demos are based on VISA (Virtual Instrument Software Architecture).

The main topics of this chapter:

- ◆ [Programming Preparations](#)
- ◆ [Excel Programming Demo](#)
- ◆ [Matlab Programming Demo](#)
- ◆ [LabVIEW Programming Demo](#)
- ◆ [Visual Basic Programming Demo](#)
- ◆ [Visual C++ Programming Demo](#)

Programming Preparations

Before programming, you need to make the following preparations:

Make sure that your PC has installed the NI-VISA library (can be downloaded from NI website: <http://www.ni.com/visa/>). Here, the default installation path is C:\Program Files\IVI Foundation\VISA.

Here, the USB interface of the oscilloscope is used to communicate with the PC and please use a USB cable to connect the USB Device interface at the rear panel of the oscilloscope to the PC. After successful connection, turn on the instrument. A "**Found New Hardware Wizard**" dialog box appears on the PC at the first connection. Please follow the instructions to install the "USB Test and Measurement Device".



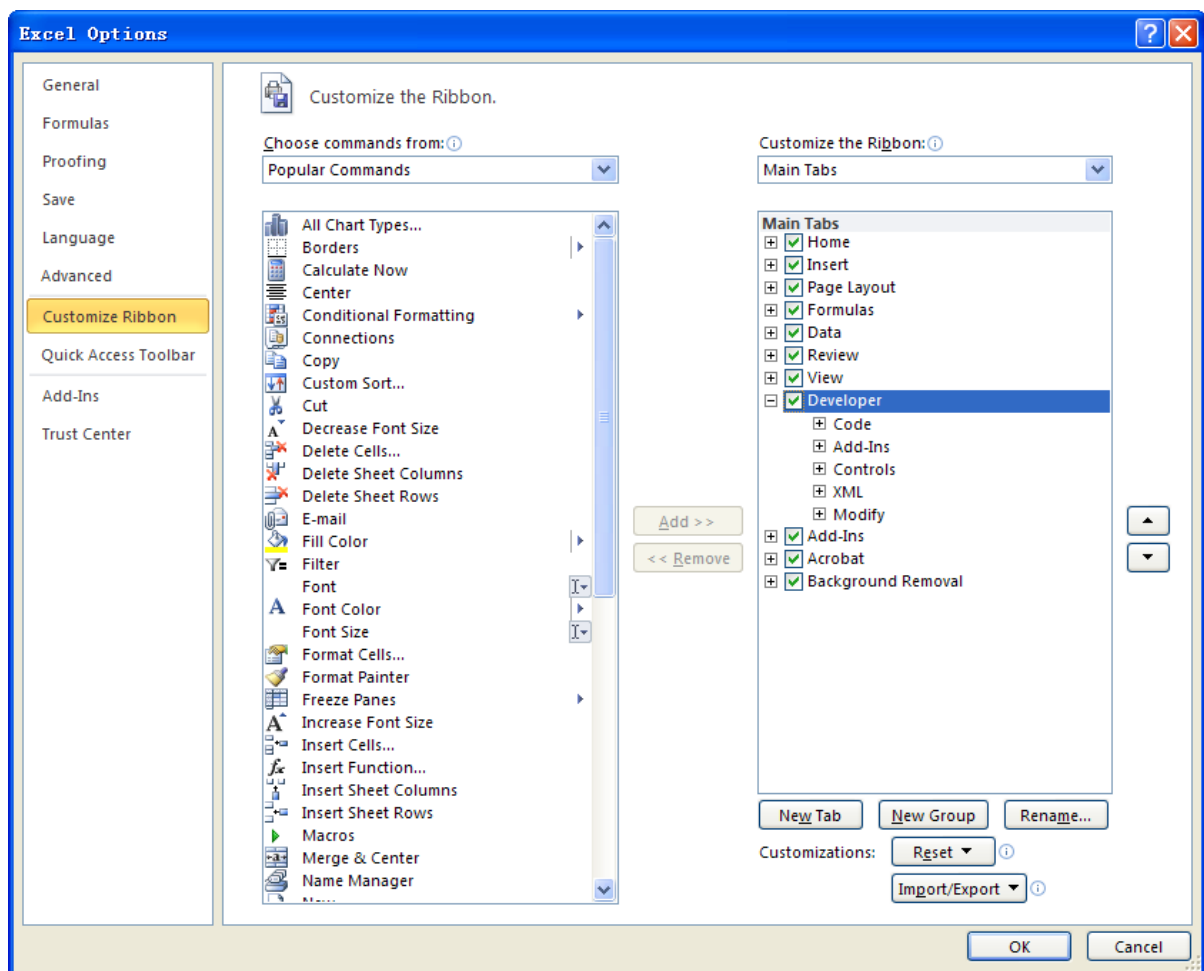
By now, the programming preparations are finished. In the following part, detailed introductions are given about the programming demos in the Excel, Labview, Matlab, Visual Basic 6.0 and Visual C++ 6.0 development environments.

Excel Programming Demo

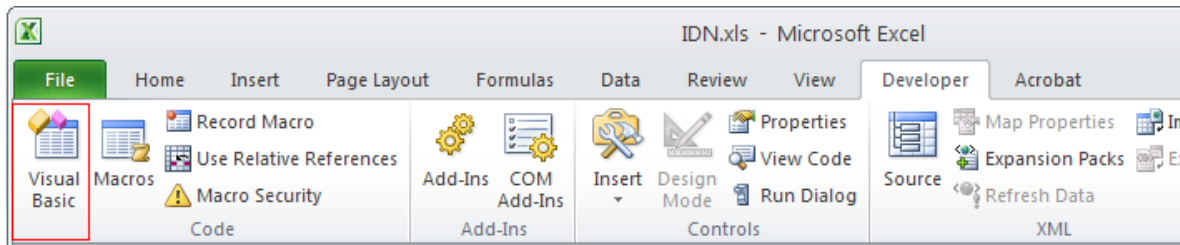
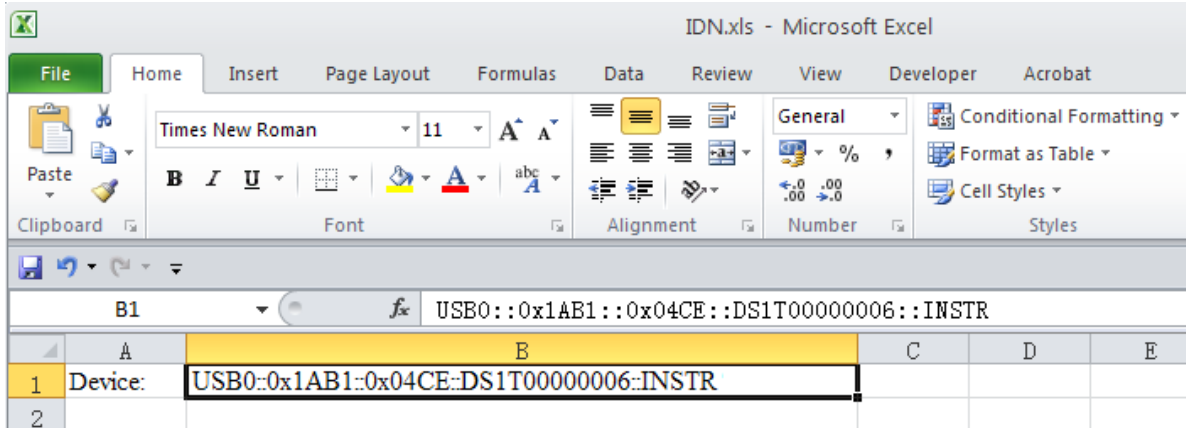
The program used in this demo: Microsoft Excel 2010

The function realized in this demo: send the *IDN? Command to read the device information.

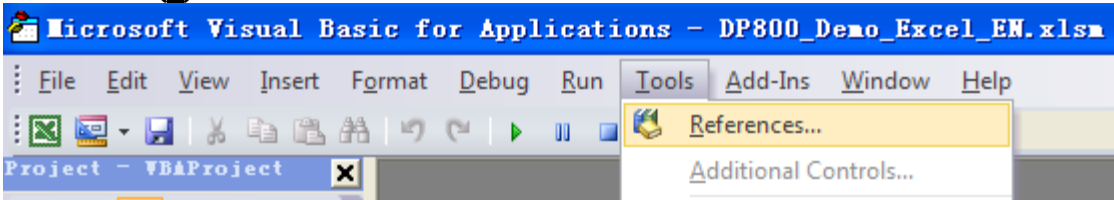
1. Create a new Excel file that enables the Macros. In this demo, the file is named as DS1000Z_Demo_Excel.xlsm.
2. Run the DS1000Z_Demo_Excel.xlsm file. Click **File**→**Options** at the upper-left corner of the Excel file to open the interface as shown in the figure below. Click **Customize Ribbon** at the right, check **Developer** and click **OK**. At this point, the Excel menu bar displays the **Developer** menu.



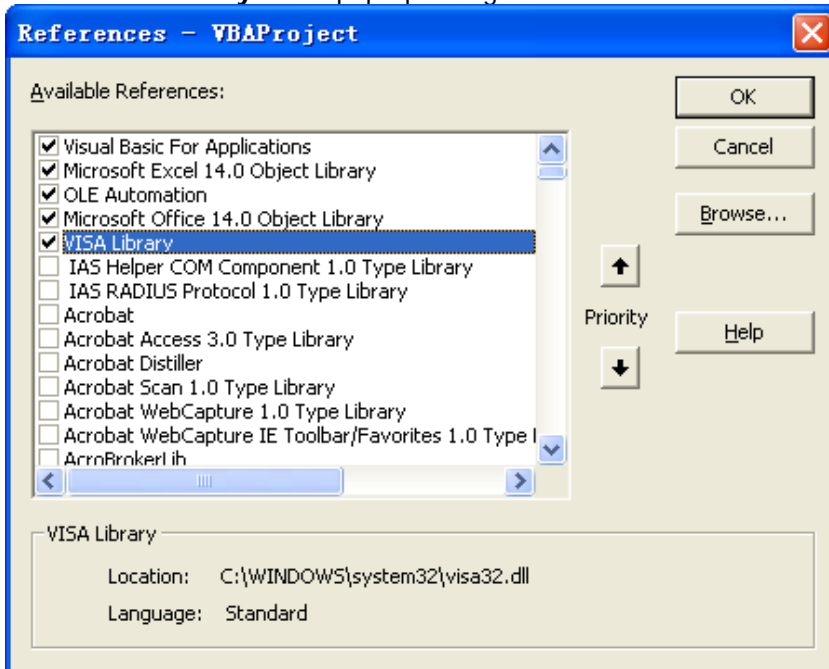
- 3. Enter the VISA descriptor into a cell of the file as shown in the figure below. For example, the VISA descriptor is USB0::0x1AB1::0x04CE::DS1T00000006::INSTR. Input it into SHEET1.CELLS(1,2) (namely the B1 cell in Sheet1). Click the **Developer** menu and select the **Visual Basic** option to open the Microsoft Visual Basic.



- 4. Select **Tools(T)** in the Microsoft Visual Basic menu bar and click **References**.



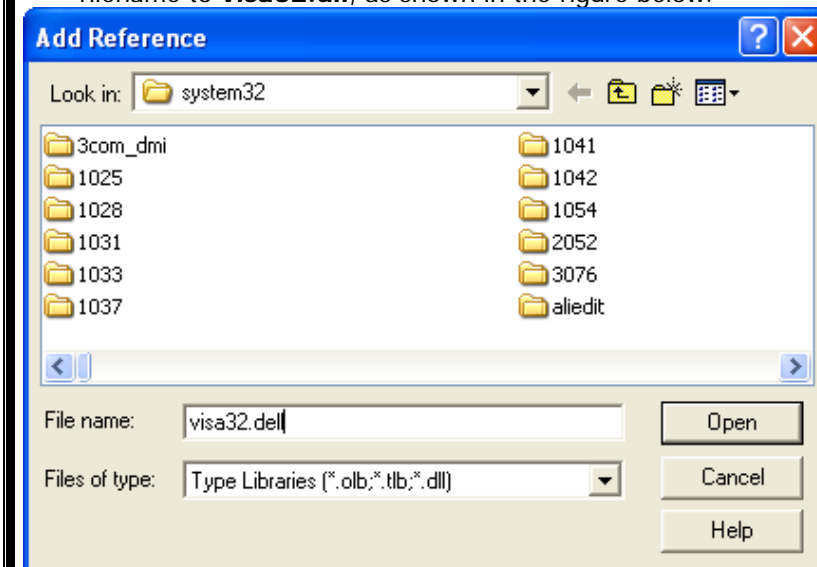
Select **VISA Library** in the pop-up dialog box and click **OK** to refer to the VISA Library.



Explanation:

If you cannot find the VISA Library in the list at the left of the figure above, please follow the method below to find it.

- (1) Make sure that your PC has installed the NI-VISA library.
- (2) Click **Browse...** at the right and set the search range to **C:\WINDOWS\system32** and the filename to **visa32.dll**, as shown in the figure below.



5. Click **View Code** in the **Developer** menu to enter the Microsoft Visual Basic interface. Add the following codes and save the file.

Note: If the Excel file created at step 2 does not enable the Macros, at this point, the prompt message "The following features cannot be saved in macro-free workbooks" will be displayed. In this situation, please save the Excel file as a file using the Macros (the filename suffix changes to .xlsm).

Sub QueryIdn()

```
Dim viDefRm As Long
Dim viDevice As Long
Dim viErr As Long
Dim cmdStr As String
Dim idnStr As String * 128
Dim ret As Long
```

'Turn on the device, the device resource descriptor is in CELLS(1,2) of SHEET1'

```
viErr = visa.viOpenDefaultRM(viDefRm)
viErr = visa.viOpen(viDefRm, Sheet1.Cells(1, 2), 0, 5000, viDevice)
```

'Send request, read the data, the return value is in CELLS(2,2) of SHEET1'

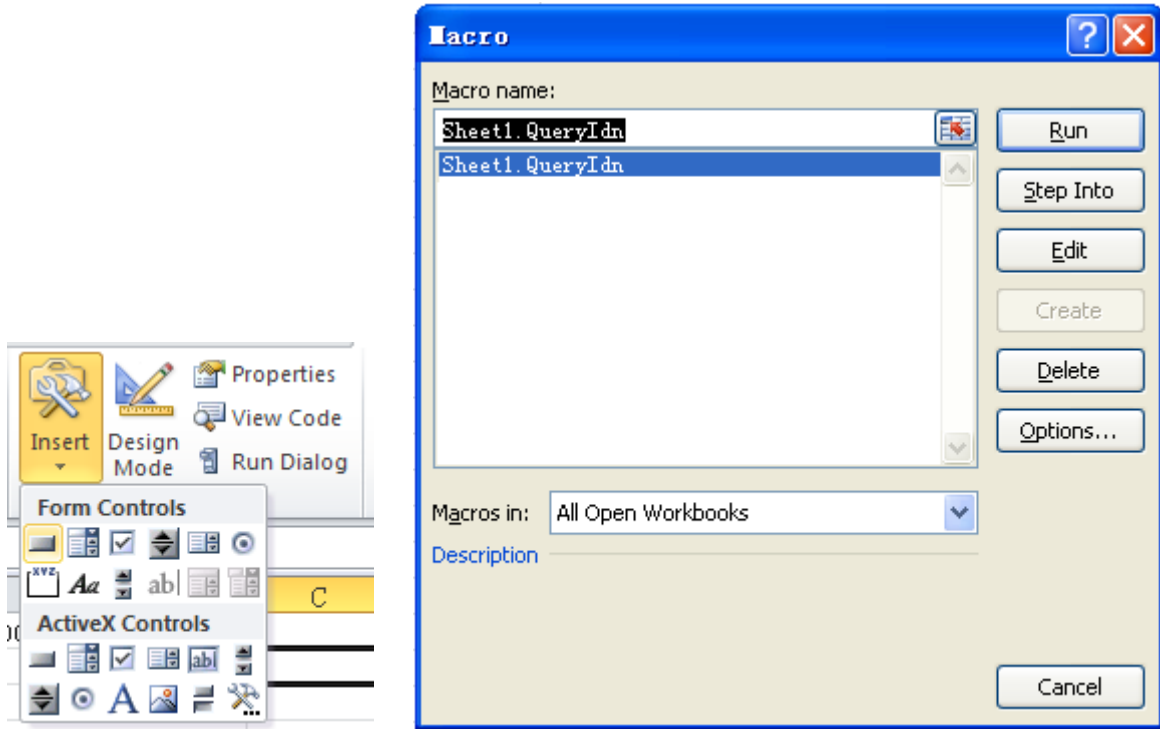
```
cmdStr = "*IDN?"
viErr = visa.viWrite(viDevice, cmdStr, Len(cmdStr), ret)
viErr = visa.viRead(viDevice, idnStr, 128, ret)
Sheet1.Cells(2, 2) = idnStr
```

'Turn off the device'

```
visa.viClose (viDevice)
visa.viClose (viDefRm)
```

End Sub

- 6. Add button control: click **Insert** in the **Developer** menu, select the desired button in **Form Controls** and put it into the cell of the Excel. At this point, the **Assign Macro** interface is displayed, select "Sheet1.QueryIdn" and click "OK".



By default, the button name is "Button 1". Right-click the button and select **Edit Text** in the pop-up menu to change the button name to "*IDN?".

- 7. Click the "*IDN?" button to send request and read data. The returned data will be displayed in SHEET1 CELLS (2,2), as shown in the figure below.

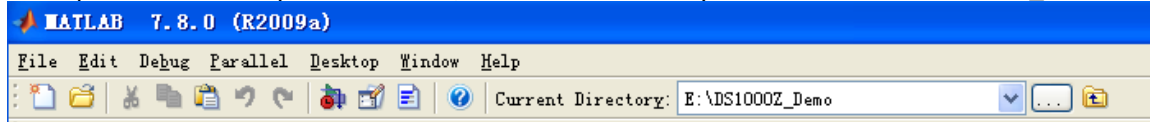
A	B	C
	USB0::0x1AB1::0x04CE::DS1T00000006:: INSTR	*IDN?
	RIGOL TECHNOLOGIES, DS1104Z, DS1T00000006, 00. 02. 00. SP1	

Matlab Programming Demo

The program used in this demo: MATLAB R2009a

The function realized in this demo: make FFT operation on the waveform data and draw the waveform.

1. Run the Matlab software and modify the current directory (namely modify the **Current Directory** at the top of the software). In this demo, the current directory is modified to E:\DS1000Z_Demo.



2. Click **File** → **New** → **Blank M-File** in the Matlab interface to create an empty M file.
3. Add the following codes in the M file:

```
% Create VISA object
DS1000z = visa('ni','USB0::0x1AB1::0x04CE::DS1T00000006::INSTR');

% Set the device property. In this demo, the length of the input buffer is set to 2048.
DS1000z.InputBufferSize = 2048;

% Open the VISA object created
fopen(DS1000z);

% Read the waveform data
fprintf(DS1000z, ':wav:data?' );

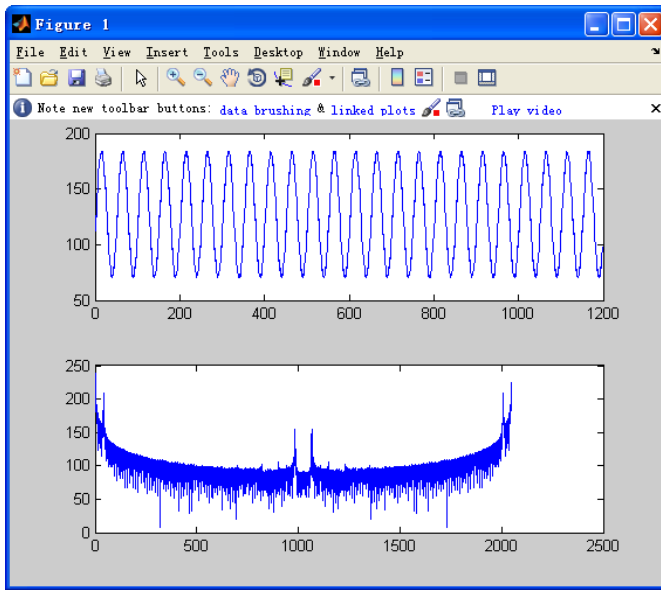
% Request the data
[data,len]= fread(DS1000z,2048);

% Close the VISA object
fclose(DS1000z);
delete(DS1000z);
clear DS1000z;

% Data processing. The waveform data read contains the TMC header. The length of the header is 11
% bytes, wherein, the first 2 bytes are the TMC header denoter (#) and the width descriptor (9)
% respectively, the 9 bytes following are the length of the data which is followed by the waveform data
% and the last byte is the terminator (0x0A). Therefore, the effective waveform points read is from the
% 12nd to the next to last.
wave = data(12:len-1);
wave = wave';
subplot(211);
plot(wave);
fftSpec = fft(wave',2048);
fftRms = abs(fftSpec);
fftLg = 20*log(fftRms);
subplot(212);
plot(fftLg);
```

4. Save the M file under the current directory. In this demo, the M file is named as DS1000Z_Demo_MATLAB.m.

5. Run the M file and the running result is as follows.

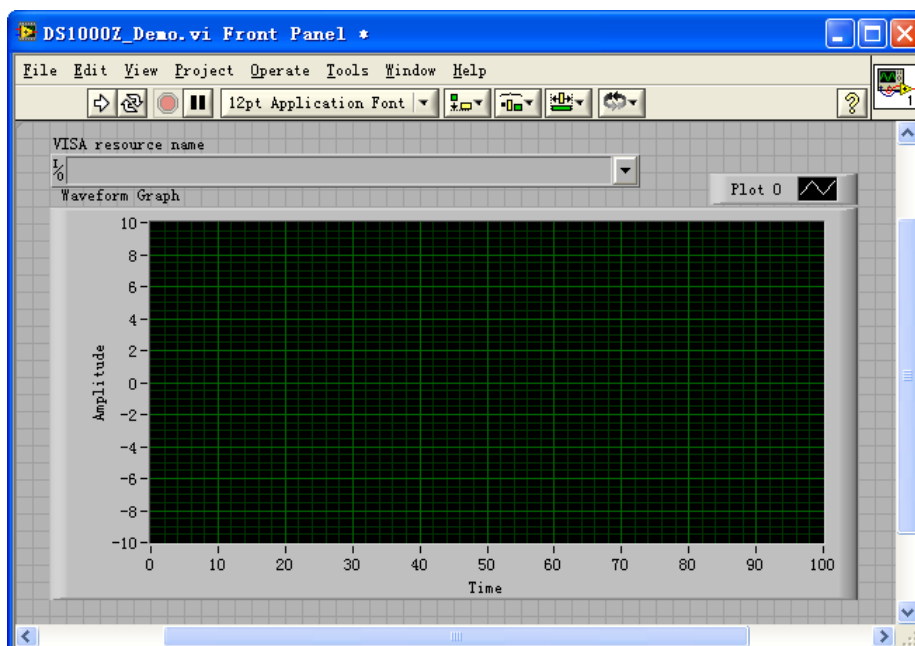
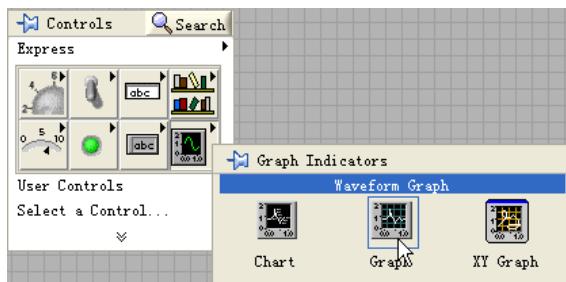
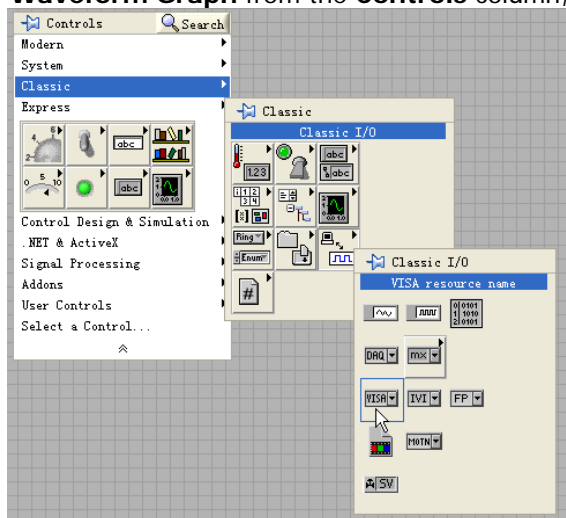


LabVIEW Programming Demo

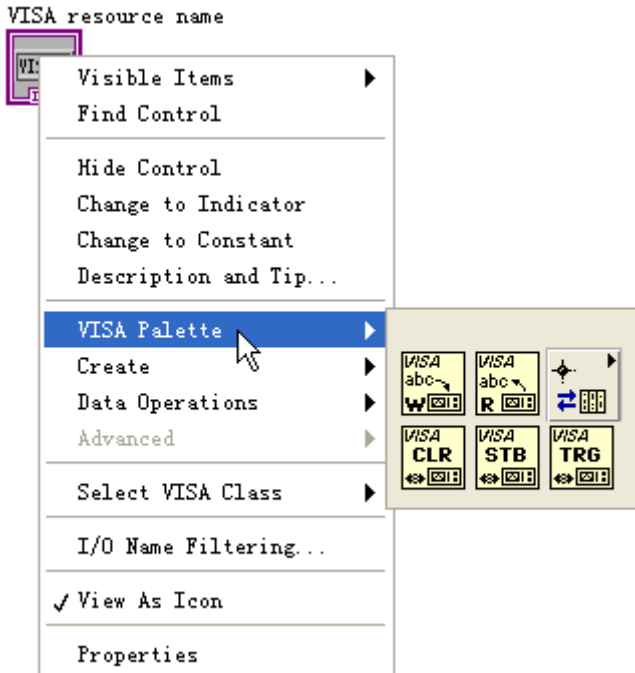
The program used in this demo: LabVIEW 2009

The functions realized in this demo: read the waveform data of CH1 on the screen.

1. Run LabVIEW 2009, create a VI file and name it as DS1000Z_Demo.
2. Add controls. Right-click in the **Front Panel** interface, select and add **VISA resource name** and **Waveform Graph** from the **Controls** column, as shown in the figures below.



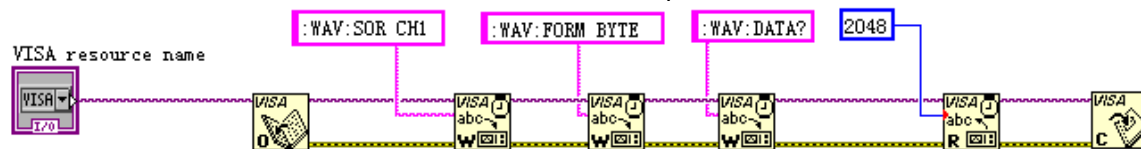
- Open the **Block Diagram** interface. Right-click on the **VISA resource name** and you can select and add the following functions from **VISA Palette** from the pop-up menu: **VISA Write**, **VISA Read**, **VISA Open** and **VISA Close**.



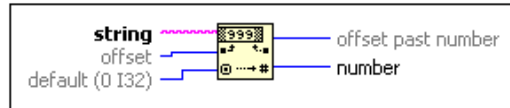
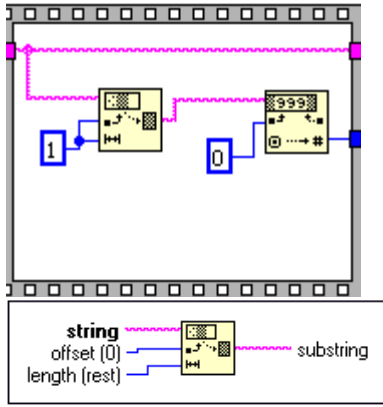
- Connect the **VISA resource name** with the **VISA Open**. Then, connect the VISA resource name outputs of all the functions with the **VISA resource name** and connect the error output with the error input as shown in the figure below.



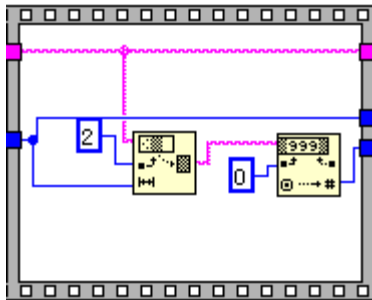
- Add text boxes in the write buffer areas of the **VISA Write** control and input `":WAV:SOR CH1"`, `":WAV:FORM BYTE"` and `":WAV:DATA?"` respectively. The first sets the channel source to CH1, the second sets the waveform reading format to BYTE and the last reads the screen waveform data. Waveform data is read through the **VISA Read** function which requires users to input the total number of bytes to be read. In this example, the total number of bytes of waveform data to be read is less than 2048. Close the VISA resource after the VISA operation is finished.



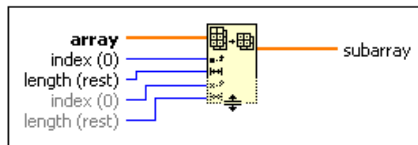
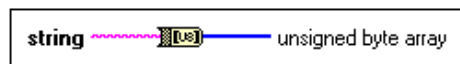
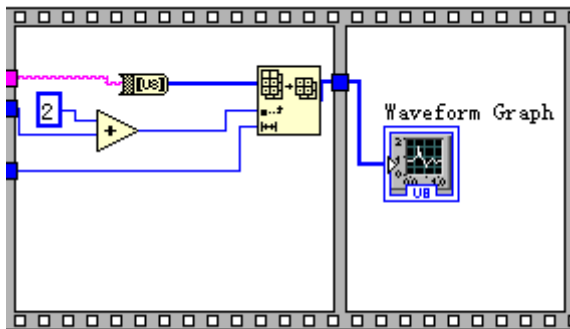
- The data format is TMC header + waveform data points + terminator. The TMC header is in `#NXXXXXX` format, wherein, `#` is the TMC header denoter, `N` represents that there are `N` bytes following. The length of the waveform data points is described in ASCII character, and the terminator represents the ending of the communication. For example, for `#9000001200XXXX`, 9 bytes are used to describe the length of the data, `000001200` represents the length of the waveform data (namely 1200 bytes). Use the **String Subset** and **Decimal String To Number** functions to get the value of `N`.



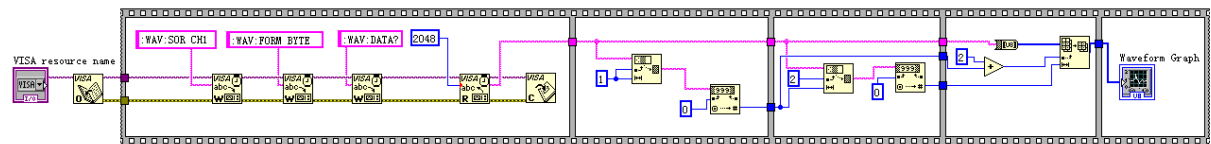
Get the length of the effective waveform data.



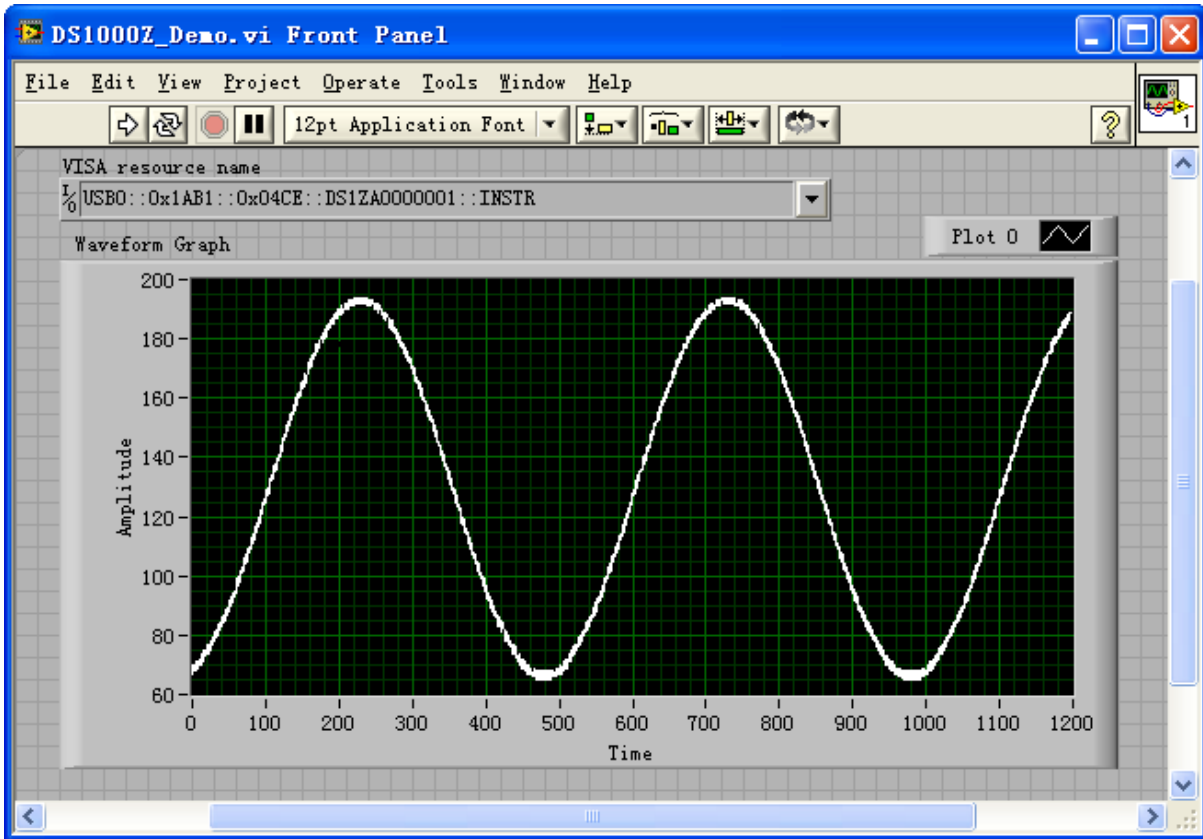
- Convert the character data into array format using the **String To Byte Array**, namely display the waveform data on the **Waveform Graph** control and then remove the TMC data header using the **Array Subset** function.



- The complete program block diagram is as shown in the figure below.



9. Select the device resource from the **VISA Resource Name** list box and run the program.



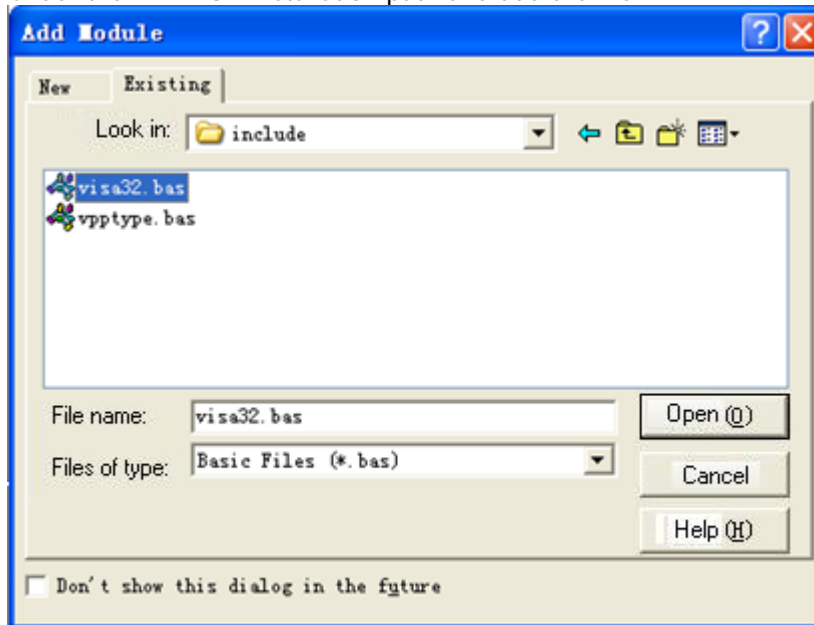
Visual Basic Programming Demo

The program used in this demo: Visual Basic 6.0

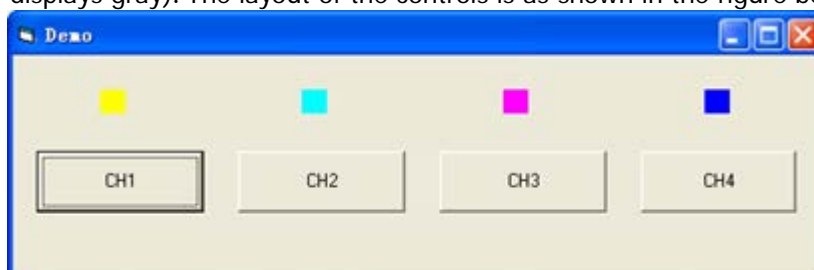
The function realized in this demo: control the on/off state of any channel.

Enter Visual Basic 6.0 and operate according to the following steps:

1. Build a standard application program project (Standard EXE) and name it as Demo.
2. Click the **Existing** tab of **Project**→**Add Module**. Search for the **visa32.bas** file in the **include** folder under the **NI-VISA** installation path and add the file.



3. Add four **CommandButton** controls to represent CH to CH4 respectively. Add four Label controls (Label1(0), Label1(1), Label1(2) and Label1(3)) to represent the status of CH1 to CH4 respectively (when the channel is enabled, it displays the color of the channel; when the channel is disabled, it displays gray). The layout of the controls is as shown in the figure below.



4. Open the **General** tab in **Project**→**Project1 Properties** and select **Form1** in the **Startup Object** dropdown box.
5. Double-click **CH1** to enter the programming environment. Add the following codes to control CH1 to CH4. The codes of CH1 are as shown below; the codes of other channels are similar.

```
Dim defrm As Long
Dim vi As Long
Dim strRes As String * 200
Dim list As Long
Dim nmatches As Long
```

```
Dim matches As String * 200 ' keep the device number acquired
Dim s32Disp As Integer
' acquire the usb source of visa
Call viOpenDefaultRM(defrm)
Call viFindRsrc(defrm, "USB?* ", list, nmatches, matches)
' Open the device
Call viOpen(defrm, matches, 0, 0, vi)
' Send the command to query the status of CH1
Call viVPrintf(vi, ":CHAN1:DISP?" + Chr$(10), 0)
' Acquire the status of CH1
Call viVScanf(vi, "%t", strRes)
s32Disp = CInt(strRes)
If (s32Disp = 1) Then
' Send the setting command
Call viVPrintf(vi, ":CHAN1:DISP 0" + Chr$(10), 0)
Label1(0).ForeColor = &H808080 'Gray
Else
Call viVPrintf(vi, ":CHAN1:DISP 1" + Chr$(10), 0)
Label1(0).ForeColor = &HFFFF& 'Yellow
End If
' Close the device
Call viClose(vi)
Call viClose(defrm)
```

6. Save and run the project and a single exe program will be obtained. When the oscilloscope is correctly connected to the PC, the ON/OFF control of any channel can be realized.

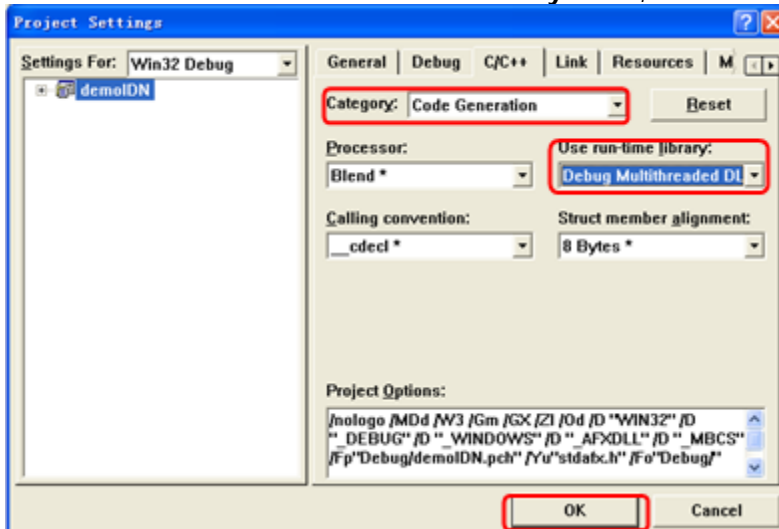
Visual C++ Programming Demo

The program used in this demo: Microsoft Visual C++ 6.0

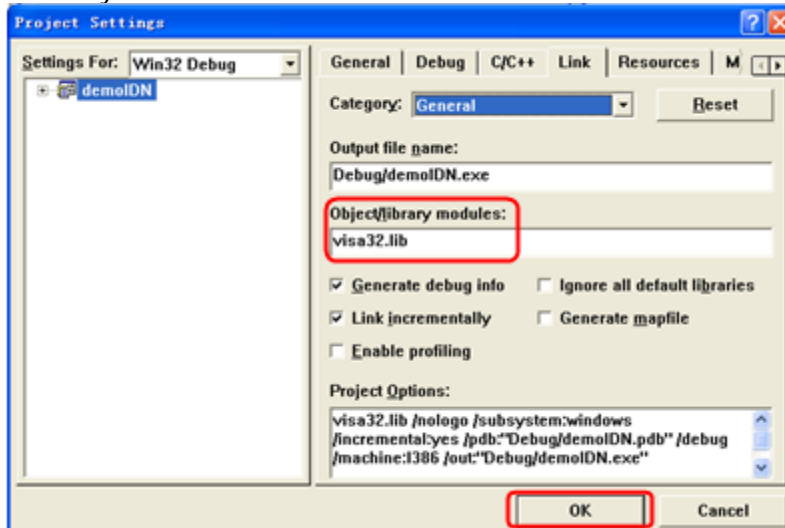
The functions realized in this demo: search for the instrument address, connect the instrument, send command and read the return value.

Enter Visual C++ 6.0 and operate according to the following steps:

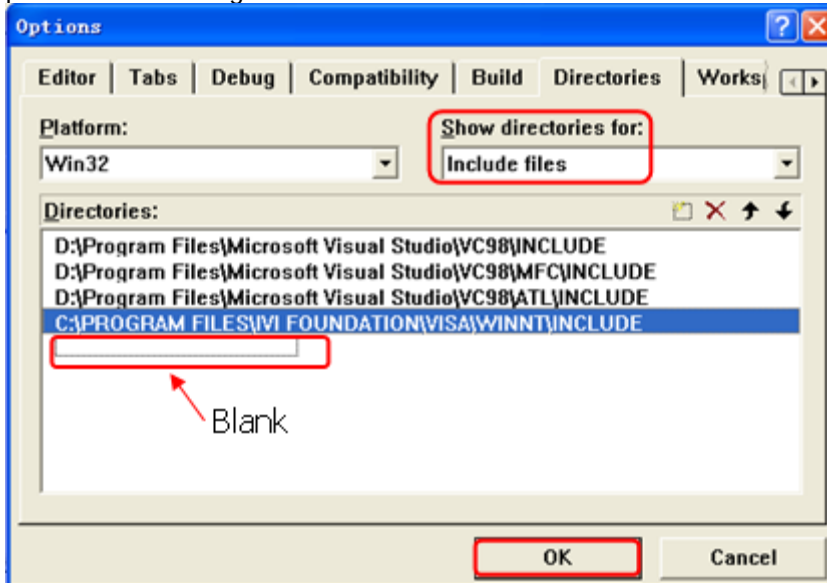
1. Create a MFC project based on dialog box.
2. Open the **C/C++** tab in **Project→Settings**, select **Code Generation** in **Category** and **Debug Multithreaded DLL** in **Use run-time library**. Then, click **OK** to close the dialog box.



3. Open the **Link** tab in **Project→Settings** and add **visa32.lib** to the **Object/library modules** manually.

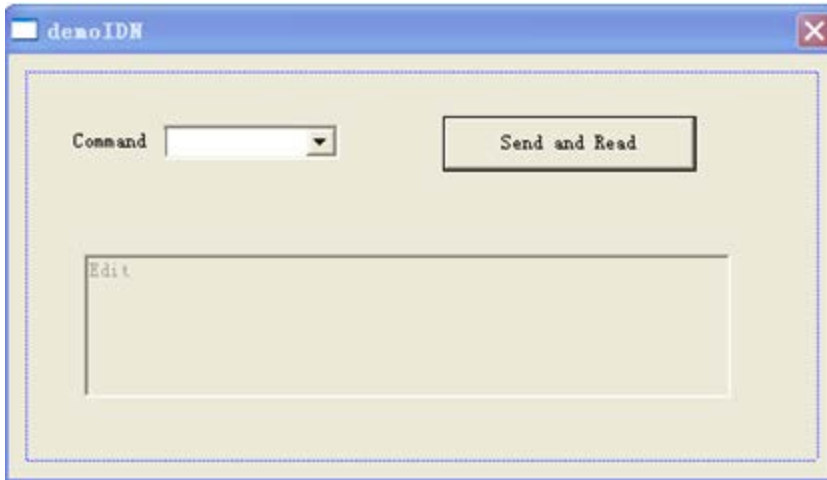


4. Open the **Directories** tab in **Tools**→**Options**.
 Select **Include files** in **Show directories for** and double-click the blank in **Directories** to add the path of **Include**: C:\Program Files\IVI Foundation\VISA\WinNT\include.
 Select **Library files** in **Show directories for** and double-click the blank in **Directories** to add the path of **Lib**: C:\Program Files\IVI Foundation\VISA\WinNT\lib\msc.



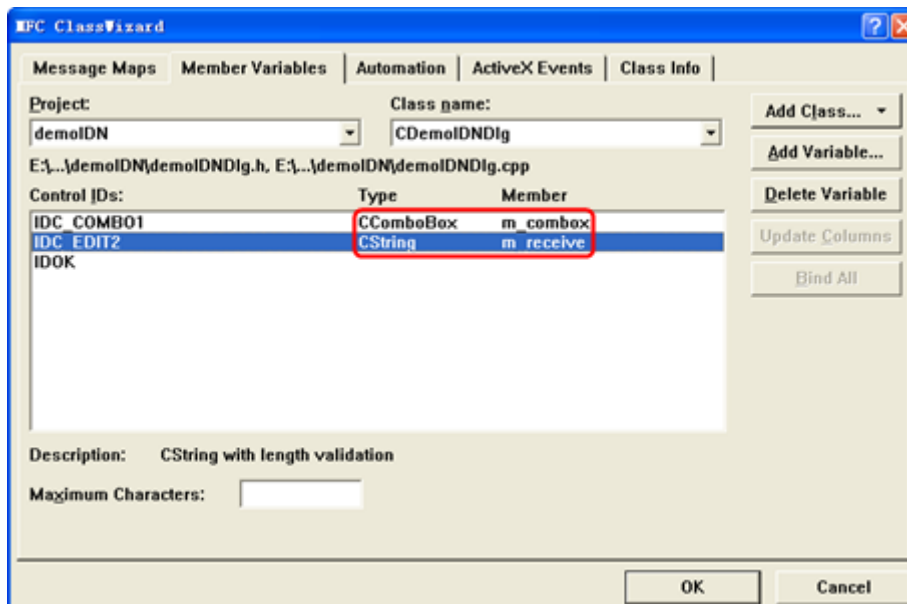
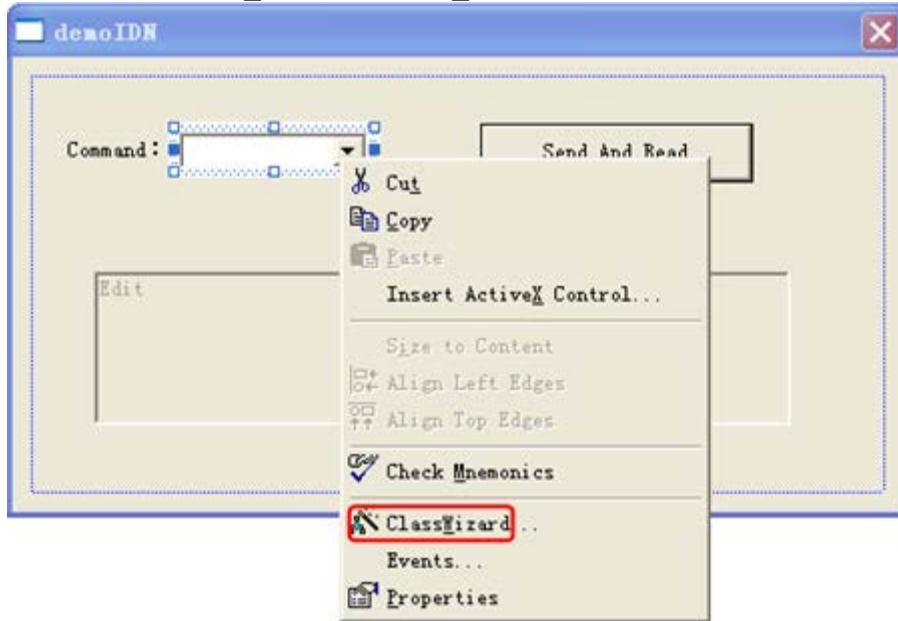
Note: at present, VISA library has been added.

5. Add the **Text**, **Combo Box**, **Button** and **Edit Box** controls as shown in the figure below.



6. Modify the control attribute.
 - 1) Name **Text** as "Command".
 - 2) Open the **Data** item in **Combo Box** attribute and input *IDN? Manually.
 - 3) Open the **General** item in **Edit Box** attribute and select **Disabled**.
 - 4) Name **Button** as **Send and Read**.

7. Add the variables **m_combox** and **m_receive** for the **Combo Box** and **Edit** controls respectively.



8. Add codes.
Double-click **Send and Read** to enter the programming environment. Declare the **#include <visa.h>** of the visa library in the header file and then add the following codes.

```
ViSession defaultRM, vi;
char buf [256] = {0};
CString s,strTemp;
char* stringTemp;
```

```
ViChar buffer [VI_FIND_BUFLLEN];
ViRsrc matches=buffer;
ViUInt32 nmatches;
ViFindList list;
```

```
viOpenDefaultRM (&defaultRM);
//Acquire the USB resource of visa
```

```
viFindRsrc(defaultRM, "USB?*") &list, &nmatches, matches);  
viOpen (defaultRM, matches, VI_NULL, VI_NULL, &vi);
```

```
// Send the command received
```

```
m_combox.GetLBText(m_combox.GetCurSel(), strTemp);  
strTemp = strTemp + "\n";  
stringTemp = (char*)(LPCTSTR)strTemp;  
viPrintf (vi, stringTemp);
```

```
// Read the result
```

```
viScanf (vi, "%t\n", &buf);
```

```
// Display the result
```

```
UpdateData (TRUE);  
m_receive = buf;  
UpdateData (FALSE);  
viClose (vi);  
viClose (defaultRM);
```

9. Save, compile and run the project and a single exe file can be obtained. When the oscilloscope is correctly connected to the PC, you can select the command *IDN? and click **Send and Read** to display the return result of the oscilloscope.

